RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

S K R I P S I

Disusun untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangka Raya

OLEH

APRIADY
NIM C1255201020
JURUSAN TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKA RAYA
2016

RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

S K R I P S I

Disusun untuk Memenuhi Syarat Kelulusan Program Strata I pada
Sekolah Tinggi Manajemen Informatika dan Komputer
(STMIK) Palangka Raya

OLEH

APRIADY
NIM C1255201020
JURUSAN TEKNIK INFORMATIKA

SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
(STMIK) PALANGKA RAYA
2016

LEMBAR PERNYATAAN

Yang bertanda tangan di bawah ini :

Nama Mahasiswa : **APRIADY**
N I M : **C1255201020**

menyatakan bahwa skripsi dengan judul :

RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

Adalah hasil karya saya dan bukan merupakan duplikasi atau seluruhnya dari karya orang lain, kecuali bagian yang sumber informasi dicantumkan.

Pernyataan ini dibuat dengan sebenar-benarnya secara sadar dan bertanggung jawab dan saya bersedia menerima sanksi pembatalan skripsi apabila terbukti melakukan duplikasi terhadap skripsi atau karya ilmiah lain yang sudah ada.

Palangka Raya, 15 Agustus 2016
Yang Membuat Pernyataan,

Materai
Rp. 6.000

APRIADY

PERSETUJUAN

RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

Skripsi ini telah disetujui untuk diujikan
pada Tanggal 02 Agustus 2016

Pembimbing I,	Pembimbing II,
Lili Rusdiana, M. Kom	Moch. Ichsan, ST, M. Kom
NIK. 198707282011007	NIK. 198001192006101

Mengetahui
Ketua STMIK Palangkaraya,

Drs. S a r t a n a, M. Si.
NIK. 195906071995102

PENGESAHAN

RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

Skripsi ini telah Diseminarkan, Dinilai dan Disahkan
Oleh Tim Seminar pada Tanggal 11 Agustus 2016

Tim Penguji Skripsi :

1. Ferdiyani Haris, M. Kom ……..
   Ketua
2. Catharina Elmayantie, M. Pd ……..
   Sekretaris
3. Rio Irawan, M. Kom ……..
   Anggota
4. Lili Rusdiana, M. Kom ……..
   Anggota
5. Moch. Ichsan, ST, M. Kom ……..
   Anggota

MOTTO DAN PERSEMBAHAN

Keberhasilan tidak datang secara tiba-tiba,

tapi karena usaha dan kerja keras.

Skripsi ini kupersembahkan untuk

- *Kedua orang tua :*
  Harmano, S. Pd dan Elnie
  untuk segala sesuatu yang
  telah diberikan
- *Saudariku :*
  Gustina Wulandari
- *Teman-temanku :*
  Kelas A Teknik Informatika
  2012 dan semua teman-
  temanku untuk semangat dan
  pendapat yang kalian berikan

ABSTRAK

Apriady, C1255201020, 2016, *Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonik HC-SR04*, Pembimbing I Lili Rusdiana, M. Kom, Pembimbing II M. Ichsan, ST, M. Kom.

Perkembangan teknologi saat ini di dunia sangat membutuhkan alat-alat atau sistem-sistem yang berfungsi secara otomatis. Permasalahan dari penelitian ini adalah bagaimana membuat rancang bangun pintu garasi otomatis menggunakan arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04. Alat ini bertujuan untuk membuat pekerjaan lebih efektif dan efisien baik dalam waktu maupun tenaga ketika membuka pintu garasi mobil. Sistem yang akan dibuat berbasis *microcontroller* menggunakan board Arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 sebagai pendeteksi mobil dan *smartphone* Android sebagai pengontrol pintu garasi.

Metode-metode penulis yang digunakan dalam penelitian ini meliputi metode *Research and Development* yaitu metode penelitian yang digunakan untuk menghasilkan produk tertentu dan metode pengumpulan data yaitu observasi, dokumentasi dan perpustakaan yang digunakan penulis untuk mengumpulkan data dengan cara membaca buku-buku, jurnal maupun artikel yang berkaitan dengan sensor ultrasonik serta tentang arduino.

Pada penelitian ini telah berhasil dilakukan pengujian *black box testing* atau pengujian kesalahan program dan hasilnya alat dan program dapat berjalan dengan baik. Berdasarkan perhitungan persentase kelompok responden sebanyak 10 responden dengan hasil persentase 88% yang berarti sistem ini layak digunakan dan di implementasikan.

Kata Kunci : Arduino UNO R3, Pintu, Garasi, Ultrasonik HC-04, Android

ABSTRACT

Apriady, C1255201020, 2016, *Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonik HC-SR04*, Advisor I Lili Rusdiana, M. Kom, Advisor II M. Ichsan, ST, M. Kom.

Current technological develpments in the world were in need of tools or systems that work automatically. The problem of this research was how to make the design of an automatic garage door using arduino UNO R3 by utilizing the HC-SR04 ultrasonic sensor. These tool aims to make the work more effective and efficient both in time and effort when opened the garage door car. The system to be created using a microcontroller-based arduino UNO R3 board by utilizing the HC-SR04 ultrasonic sensor as a detector of the car.

The methods used in this study included the Research and Developmenty method to produce certain products and the data collection method used by the writer was observation, documentation and library method it was to gether data by reading books, journals and articles releated to ultrasonic sensor and about arduino.

This study had been successfully carried out tested black-box testing or tested errors program and the results are tools and programs could be run properly. Based on the caclucation of the perecentage of the group of as many as 10 respondents with the percentage of 88% which means the system was fit for used and implemented.

Keywords : Arduino UNO R3, Doors, Garage, Ultrasonic HC-04, Android

KATA PENGANTAR

Dengan memajatkan puji dan syukur kehadirat Tuhan Yang Maha Esa, atas berkat dan rahmat-Nya sehingga penulis dapat menyelesaikan skripsi dengan judul "**RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04"**.

Pada kesempatan ini penulis juga menyampaikan banyak terima kasih yang sebesar besarnya kepada semua pihak yang telah memberikan bimbingan dan motivasi dalam penyelesaian penulisan skripsi ini, yaitu kepada :

1. Drs. Sartana, M. Si selaku Ketua Sekolah Tinggi Manajemen Informatika dan Komputer (STMIK) Palangkaraya.
2. Lili Rusdiana, M. Kom selaku dosen pembimbing I yang banyak memberikan saran, koreksi dan bimbingan dalam menyelesaikan skripsi ini.
3. M. Ichsan, ST, M. Kom selaku dosen pembimbing II yang banyak memberikan saran, koreksi, pengetahuan dan bimbingan dalam menyelesaikan skripsi ini.
4. Kedua orang tua tercinta Ibu dan Bapak, saudara, keluarga dan teman-teman yang telah memberikan dukungan selama penyusunan dan penulisan skripsi ini.

Akhir kata penulis mengharapkan kritik dan saran yang membangun untuk penyempurnaan penulisan skripsi ini.

Palangkaraya, 15 Agustus 2016

Penulis,

DAFTAR ISI

HALAMAN SAMPUL ii
HALAMAN PERNYATAAN iii
HALAMAN PERSETUJUAN iv
HALAMAN PENGESAHAN v
HALAMAN MOTTO DAN PERSEMBAHAN vi
HALAMAN ABSTRAK vii
HALAMAN ABSTRACT viii
KATA PENGANTAR ix
DAFTAR ISI xi
DAFTAR TABEL xiv
DAFTAR GAMBAR xv
DAFTAR LAMPIRAN xvi

BAB I PENDAHULUAN
- A. Latar Belakang Masalah 1
- B. Perumusan Masalah 2
- C. Batasan Masalah 2
- D. Tujuan dan Manfaat 3
  - 1. Tujuan 3
  - 2. Manfaat 3
    - a. Bagi penulis 3
    - b. Bagi STMIK Palangkaraya 3
    - c. Bagi masyarakat 4
- E. Metode Penelitian 4
  - 1. Teknik Pengumpulan Data 4
    - a. Kepustakaan 4
    - b. Observasi 5
    - c. Dokumentasi 5
  - 2. Tahap Pengembangan Sistem 5
- F. Sistematika Penulisan 5

BAB II LANDASAN TEORI
- A. Tinjauan Pustaka 7
- B. Kajian Teori 10
  - 1. Rancang Bangun 10
  - 2. Pintu 10
  - 3. Garasi 11
  - 4. Otomatis 11
  - 5. Arduino 11
  - 6. Sensor Ultrasonik HC-SR04 15
  - 7. Bluetooth HC-05 16
  - 8. Motor servo 18
  - 9. LED 19

10. Resistor 20
11. *Breadboard* 20
12. Model *Prototype* 21
13. Bagan Alir Sistem atau Dokumen (*Flowchart*) 23
14. *Black-Box Testing* 24
C. Perangkat Lunak yang Digunakan 25
1. Arduino IDE 25
2. MIT App Inventor 2 beta 27

BAB III ANALISA DAN DESAIN SISTEM
A. Tinjauan Umum 33
B. Analisis..... 33
1. Analisis Kelemahan Sistem 33
2. Analisis Kebutuhan Sistem..... 34
a. Kebutuhan Perangkat Keras 34
b. Kebutuhan Perangkat Lunak 35
c. Kebutuhan Informasi 35
d. Kebutuhan pengguna 35
3. Analisis Kelayakan Sistem 36
a. Kelayakan Teknologi 36
b. Kelayakan Hukum 36
c. Kelayakan Operasinal 36
C. Desain Sistem 37
1. Desain Proses 37
a. Diagram Blok 37
b. Rangkaian Sistem 38
c. *Flowchart* Program Aplikasi 39
2. Desain *Inteface* / Antarmuka 41

BAB IV IMPLEMENTASI DAN PEMBAHASAN
A. Implementasi 44
1. Uji Coba Program 44
2. Implementasi Sistem 45
3. Panduan Program 46
4. Panduan Instalasi 46
a. Panduan Instalasi Arduino 46
b. Panduan Instalasi Aplikasi 51
B. Pembahasan 52
1. Pembahasan *Source code* Program Arduino 52
2. Pembahasan *Blocks* Program MIT App Inventor 55
3. Pembahasan *Interface* / Antarmuka Program 59
a. Tampilan Utama 59
b. Tampilan Tentang 60
c. Tampilan Bantuan 61
4. Pembahasan Hasil Respon Pengguna 61
a. Kuisioner 62

b. Hasil Kuisioner 63

BAB V KESIMPULAN
A. Kesimpulan 65
B. Saran 66

DAFTAR PUSTAKA
LAMPIRAN

DAFTAR GAMBAR

Gambar 1. Logo Arduino 12
Gambar 2. Arduino UNO 13
Gambar 3. Sensor Ultrasonik HC-SR04 15
Gambar 4. Prinsip Kerja Sensor Ultrasonik 16
Gambar 5. Bluetooth HC-05 17
Gambar 6. Motor Servo SG90 9g 18
Gambar 7. LED 19
Gambar 8. Resistor 20
Gambar 9. Boardboard 21
Gambar 10. Model *Prototype* 22
Gambar 11. Arduino IDE 25
Gambar 12. Contoh *Blocks* 28
Gambar 13. Jendela Desainer 28
Gambar 14. *Blocks Editor* 30
Gambar 15. *Emulator* 31
Gambar 16. Diagram Blok 37
Gambar 17. Rangkaian Sistem 38
Gambar 18. *Flowchart* Program Aplikasi 40
Gambar 19. Rancangan Tampilan Utama 41
Gambar 20. Rancangan Tampilan Tentang 42
Gambar 21. Rancangan Tampilan Bantuan 43
Gambar 22. Pintu Garasi Otomatis 45
Gambar 23. *Sketch* Arduino IDE 47
Gambar 24. *Port* Arduino UNO 47
Gambar 25. *Setting Board* Arduino IDE 48
Gambar 26. *Setting Port* Arduino IDE 49
Gambar 27. Proses *compile* Arduino IDE 50
Gambar 28. Proses *upload* Arduino IDE 50
Gambar 29. Proses Instalasi Aplikasi *Controller* 51
Gambar 30. Proses Instalasi Selesai 52
Gambar 31. *Blocks* Tombol *Connect* 56
Gambar 32. *Blocks* Tombol *Disconnect* 56
Gambar 33. *Blocks* Tombol *Unlock* 57
Gambar 34. *Blocks* Tombol *Lock* 57
Gambar 35. *Blocks* Tombol *Garage Door Controller* 57
Gambar 36. *Blocks* Tombol *Exit* 58
Gambar 37. *Blocks* Tombol *Help* 58
Gambar 38. *Blocks* Tombol *Back* 58
Gambar 39. Tampilan Utama 59
Gambar 40. Tampilan Tentang 60
Gambar 41. Tampilan Bantuan 61
Gambar 42. Persentase Hasil Kuisioner 64

DAFTAR TABEL

Tabel 1. Tabel Simbol-Simbol *Flowchart* 23
Tabel 2. Uji Coba Pintu Garasi Otomatis 44
Tabel 3. Kuisioner 62
Tabel 4. Hasil Kuisioner 63

DAFTAR LAMPIRAN

Lampiran 1. Surat Tugas

Lampiran 2. Kartu Kegiatan Konsultasi Skripsi

Lampiran 3. Berita Acara Ujian Skripsi

Lampiran 4. Berita Acara Eksperimen Program

Lampiran 5. Kuisioner

Lampiran 6. Prototype 1 Pembuatan Garasi

Lampiran 7. Prototype 2 Pembuatan Tempat Rangkaian Sistem Komponen

Lampiran 8. Prototype 3 Pemasangan Pintu Garasi, LED dan Sensor Ultrasonik

Lampiran 9. Prototype 4 Pemasangan Sensor Ultrasonik dan Sistem Komponen

Lampiran 10. Prototype 5 Pemasangan Garasi dan Tempat Sistem Komponen

Lampiran 11. Prototype 6 Pemasangan Servo dan Sistem Penggerak Pintu Garasi

BAB I
PENDAHULUAN

A. Latar Belakang Masalah

Perkembangan ilmu pengetahuan dan teknologi yang semakin maju banyak dimanfaatkan manusia untuk memenuhi kebutuhan sehari-hari. Perkembangan teknologi yang pesat ini ditandai dengan banyaknya peralatan yang telah diciptakan dan dioperasikan baik secara manual maupun otomatis. Berkat perkembangan teknologi saat ini penulis ingin membuat efisiensi atau kemudahan didalam beraktivitas, maka diharapkan dengan cara ini dapat menjadi solusi dan menjawab keluhan dari masyarakat tentang pemanfaatan teknologi untuk kebutuhan kecil.

Perkembangan teknologi saat ini di dunia sangat membutuhkan alat-alat atau sistem-sistem yang berfungsi secara otomatis. Hal ini berguna untuk membuat pekerjaan lebih efektif dan efisien baik dalam waktu maupun tenaga. Untuk memenuhi kebutuhan otomatisasi ini diperlukan peralatan yang bisa memenuhi kebutuhan tersebut, diantaranya alat kontrol berbasis *microcontroller* Arduino UNO R3, sensor ultrasonik HC-SR04 dan *smartphone* Android yang dapat digunakan untuk otomatisasi pada pintu garasi.

Dalam kehidupan sehari-hari banyak hal yang dilakukan di dalam dan di luar ruangan, bahkan aktifitas tersebut tidak lepas dari penggunaan pintu, seperti pintu garasi yang harus dibuka dan ditutup secara berulang-ulang kali. Pintu yang terpasang umumnya susah bergerak, sehingga kurang praktis

penggunaannya. Melihat kondisi yang ada kebanyakan proses pengoperasian pintu garasi mobil masih dilakukan secara manual dengan cara mengangkat atau menarik pintu garasi tersebut, bagi sebagian orang kegiatan seperti itu mungkin saja menjadi sebuah hal yang cukup merepotkan.

Berawal dari kondisi tersebut, maka penulis bermaksud untuk membuat simulasi *"Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino Uno R3 dengan Memanfaatkan Sensor Ultrasonik HC-SR04"*.

B. Perumusan Masalah

Berdasarkan latar belakang di atas, maka didapatkan rumusan masalah yaitu "bagaimana membuat rancang bangun pintu garasi otomatis menggunakan arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04?"

C. Batasan Masalah

Agar permasalahan dalam penelitian ini tidak melebar luas, maka diberikan batasan-batasan untuk pendekatan permasalahan agar lebih terinci dalam pelaksanaannya, yaitu:

1. *Board* Arduino yang digunakan adalah Arduino UNO R3.
2. Sensor ultrasonik HC-SR04 digunakan sebagai indikator objek.
3. Motor penggerak pintu garasi menggunakan motor servo.
4. Pengunci pintu hanya menggunakan kondisi motor yang tidak bergerak.
5. Sistem keamanan pintu garasi menggunakan aplikasi *controller* pada *smartphone* Android.

6. Daya yang dibutuhkan untuk sistem 9V-12V tanpa daya tambahan.
7. Simulasi menggunakan *prototype* garasi dan mobil RC.
8. Sistem yang dibuat hanya sebagai simulasi dari kejadian sebenarnya.

D. Tujuan dan Manfaat

1. Tujuan

   Tujuan penelitian ini adalah untuk merancang sebuah pintu garasi otomatis menggunakan arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04.

2. Manfaat

   Penelitian ini diharapkan dapat memberikan manfaat, yaitu:

   a. Bagi penulis

      Adapun manfaat penelitian ini adalah untuk mengimplementasikan ilmu yang didapat pada saat perkuliahan dan mengembangkan ilmu dibidang elektronika.

   b. Bagi STMIK Palangkaraya

      Manfaat yang diberikan kepada kampus adalah sebagai penambah literatur pustaka pada perpustakaan STMIK Palangkaraya serta dapat digunakan sebagai referensi dan dokumen akademik yang berguna untuk dijadikan acuan bagi civitas akademik di STMIK Palangkaraya.

c. Bagi masyarakat

Adapun manfaat penelitian bagi masyarakat yaitu diharapkan dapat menjadi acuan agar dapat diterapkan pada kondisi sebenarnya agar membuat pekerjaan lebih efektif dan efisien baik dalam waktu maupun tenaga untuk membuka atau menutup pintu garasi mobil tanpa harus mendorong ataupun menarik pintu garasi.

E. Metode Penelitian

Dalam penelitian ini penulis menggunakan metode *Research and Development* atau R&D. Metode penelitian *Research and Development* yang disingkat R&D adalah metode penelitian yang digunakan untuk menghasilkan produk tertentu, dan menguji keefektifan produk tersebut (Sugiyono:2009:407).

Adapun tahapan yang dilakukan penulis dalam melakukan penelitian adalah sebagai berikut :

1. Teknik Pengumpulan Data

a. Kepustakaan

Kepustakaan yaitu metode yang dilakukan dengan cara membaca buku-buku yang relevan dengan apa yang peneliti lakukan baik itu dari perpustakaan atau sumber lain. Seperti buku-buku, jurnal maupun artikel yang membahas tentang sensor ultrasonik serta tentang Arduino.

b. Observasi

Observasi yaitu metode pengumpulan data dengan cara melakukan pengamatan terhadap fungsi dari komponen-komponen yang akan digunakan dalam merancang *prototype* garasi.

c. Dokumentasi

Metode ini dilakukan dengan mengumpulkan data-data yang berhubungan dengan tahapan-tahapan pembuatan *prototype* garasi.

2. Tahap Pengembangan Sistem

Dalam penelitian ini penulis menggunakan model pengembangan perangkat yaitu *Prototype*.

F. Sistematika Penulisan

Pada penulisan skripsi ini, penulis membuat sistematika penulisan agar memudahkan dalam membaca dan memahami isi dari skripsi ini secara garis besarnya dengan membagi menjadi beberapa bab, sebagai berikut:

BAB I PENDAHULUAN

Pada bab ini berisi tentang latar belakang masalah, perumusan masalah, batasan masalah, tujuan dan manfaat, metode penelitian dan sistematika penulisan.

BAB II LANDASAN TEORI

Pada bab ini berisi tentang definisi, karakteristik sistem, konsep pemodelan sistem dan perangkat lunak yang digunakan.

BAB III ANALISA DAN DESAIN SISTEM

Pada bab ini berisi tentang tinjauan umum objek penelitian, analisis kelemahan sistem, analisis kebutuhan, analisis kelayakan, desain proses dan desain *interface*.

BAB IV IMPLEMENTASI DAN PEMBAHASAN

Pada bab ini berisi tentang implementasi alat dengan uji program, uji sistem, manual program, pemeliharaan sistem, pembahasan listing program, pembahasan *interface* dan pembahasan hasil respon pengguna.

BAB V KESIMPULAN DAN SARAN

Pada bab ini berisi tentang kesimpulan yang didapat dari pembahasan pada bab-bab sebelumnya dan saran-saran yang dianggap perlu diperhatikan sehubungan dengan implementasi alat dan pengembangan untuk kedepannya.

BAB II
LANDASAN TEORI

A. Tinjauan Pustaka

Tinjauan pustaka yang digunakan penulis berasal dari sumber yang relevan dengan penelitian yang sedang penulis lakukan.

1. **Penelitian yang dilakukan oleh Mohammad Thoif (2006) dengan judul "Pengendali Pintu Gerbang dan Garasi Secara Otomatis" Tugas Akhir Fakultas Teknik Universitas Negeri Semarang, di Semarang** hasil penelitiannya adalah membuat Pintu Gerbang dan Garasi Otomatis dengan menggunakan IC TX-2 sebagai IC dan Motor DC 12 V 2 buah untuk membuka pintu gerbang dan *remote control* sebagai pengendali pintu gerbang dan garasi, dan Osilator RF sebagai pembangkit sinyal dan motor listrik sebagai penggerak untuk membuka dan menutup gerbang, dengan cara pemancar akan mengirimkan gelombang frekuensi tinggi / RF ke antena penerima gelombang sinyal untuk menggerakan motor listrik yang membuka atau menutup gerbang. Selanjutnya oleh rangkaian penerima gelombang RF dan sinyal asli akan dipisahkan, gelombang RF dibuang dan sinyal asli diambil untuk mengendalikan motor listrik.
2. **Penelitian yang dilakukan oleh Brata Abi Mantra (2006) dengan judul "Simulasi Pintu Garasi Mobil Otomatis Berbasis PLC (*Programmable Logic Control*)" Tugas Akhir Fakultas Teknik Universitas Semarang di Semarang** hasil penelitiannya adalah membuat Simulasi Pintu Garasi

Otomatis menggunakan PLC sebagai otak pengendali untuk menggerakan pintu garasi dengan cara memberi masukan PLC menggunakan empat pasang dioda sinar laser dan LDR (*Light Dipendent Resistor*) yaitu 00.001-00.004, yang akan bekerja apabila resistensi dari LDR yang terkena cahaya oleh dioda laser tersebut berkurang yang kemudian dihubungkan dengan relai sebagai masukan pada PLC serta menggunakan dua buah *limit switch* (00.005-00.006), untuk menghentikan putaran motor. Sebagai aktuatornya (keluaran) menggunakan sebuah motor DC 12 V (01.001 dan 01.002), yang digunakan untuk memutar *rolling door* untuk membuka ataupun menutup pintu.

3. **Penelitian yang dilakukan oleh Tubagus Reza (2007) dengan judul "Pintu Geser Otomatis" Skripsi Universitas Gunadarma, di Depok** hasil penelitiannya adalah membuat Pintu Geser dengan menggunakan *Microcontroller* Seri AT89, sensor *Infrared* dan motor stepper dengan cara apabila sensor *Infrared* dan photo dioda yang saling berhadapan tidak dihalangi oleh suatu benda maka nilai logika masih bernilai 1, maka IC tidak akan memerintahkan motor *stepper* akan berputar. Namun apabila sensor *Infrared* dan photo dioda terhalang oleh suatu benda, maka nilai logika photo dioda akan menjadi nol, seperti yang di ketahui sensor photo dioda merupakan sensor yang sensitif terhadap cahaya, apabila photo dioda mendapatkan sinar maka photo dioda akan mengalirkan arus dan

sebaliknya apabila tidak mendapat sinar khususnya sinar *Infrared* maka sensor photo dioda tidak mengalirkan arus dan arus masuk melalui rangkaian comparator. Lalu IC akan memerintahkan untuk mengaktifkan *driver* dan motor *stepper* untuk berputar searah jarum jam dan menggerakan kerangka pintu untuk menggeser ke kanan. Setelah itu ada jeda waktu (delay) selama 5 menit lalu pintu akan menutup kembali dan melakukan pengecekan kembali apakah sensor terhalang atau tidak.

4. **Penelitian yang dilakukan oleh Hendra Maryanto (2010) dengan judul "Pembuatan Prototipe Pintu Otomatis Satu Arah Berbasis *Microcontroller* ATMega 8535 Menggunakan Double IR" Tugas Akhir Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret Surakarta, di Surakarta** hasil penelitiannya adalah merancang prototipe pintu otomatis menggunakan sensor PIR, *microcontroller* ATMega 8535, IC L293D dan Motor DC. Kondisi pertama adalah kedua PIR yaitu PIR 1 dan PIR 2 dalam keadaan normal dan pintu dalam keadaan tertutup, setelah PIR 1 mendeteksi suhu tubuh manusia maka LED indikator pada PIR 1 akan menyala dan *vout* akan berlogika 0, *vout* yang terhubung dengan portc.6 akan memberi perintah pada *microcontroller* yang kemudian diteruskan ke IC L293D untuk menggerakan motor DC yang berakibat pintu dapat terbuka secara otomatis. Kondisi kedua adalah setelah pintu terbuka maka sensor PIR 2

yang masih dalam keadaan normal akan mendeteksi suhu tubuh manusia maka LED indikator pada PIR 2 akan menyala dan *vout* akan berlogika 0. *Vout* yang terhubung dengan portc.7 akan memberi perintah pada *microcontroller* yang kemudian diteruskan ke IC L293D untuk menggerakkan motor DC yang berakibat pintu dapat tertutup kembali secara otomatis. Singkatnya *microcontroller* menerima input dari sensor PIR, kemudian *microcontroller* memberi output kepada IC L293D, selanjutnya keluaran dari IC L293D masuk ke motor DC yang berfungsi untuk membuka dan menutup pintu.

B. Kajian Teori

Kajian teori yang penulis gunakan berasal dari buku-buku yang relevan dengan penelitian yang sedang penulis lakukan.

1. Rancang Bangun

Rancang Bangun (*desain*) adalah tahap dari setelah analisis dari siklus pengembangan sistem yang merupakan pendefinisian dari kebutuhan-kebutuhan fungsional, serta menggambarkan bagaimana suatu sistem dibentuk yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah ke dalam satu kesatuan yang utuh dan berfungsi, termasuk menyangkut mengkonfigurasikan dari komponen-komponen perangkat keras dan perangkat lunak dari suatu sistem (Jogiyanto 2013:2).

2. Pintu

Pintu digunakan untuk jalan keluar masuk baik manusia maupun kendaraan dan berfungsi sebagai pengaman suatu ruang yang menghubungkan area luar dan dalam suatu ruang (Dzilfikar, 2011:3).

3. Garasi

Garasi adalah tempat menyimpan perkakas, suku cadang kendaraan ataupun tempat untuk menyimpan mobil agar terhindar dari pencurian ataupun melindungi mobil terhadap cuaca terik matahari dan air hujan (Dzilfikar, 2011:15).

4. Otomatis

Otomatis adalah suatu teknologi terkait dengan mekanik, elektronik, dan komputer-komputer didasarkan sistem untuk beroperasi dengan sendirinya (Saftari, 2015:17).

5. Arduino

Arduino adalah papan rangkaian elektronik (*electronic board*) *open source* yang didalamnya terdapat komponen utama yaitu sebuah chip *microcontroller* (Saftari, 2015:1).

Arduino adalah nama keluarga papan *microcontroller* yang awalnya dibuat oleh perusahaan Smart Project. Salah satu tokoh penciptanya adalah Massimo Banzi. Papan ini merupakan perangkat keras

yang bersifat "*open source*" sehingga boleh dibuat siapa saja. Logo Arduino seperti pada gambar 1.

Gambar 1. Logo Arduino

Arduino dibuat dengan tujuan untuk memudahkan *eksperimen* atau perwujudan berbagai peralatan yang berbasis *microcontroller*, misalnya:

a. Pelacakan lokasi mobil

b. Penyiraman tanaman secara otomatis

c. Otomatisasi akses pintu ruangan

d. Pendeteksi keberadaan orang untuk pengambilan keputusan

Berbagai jenis kartu Arduino tersedia, antara lain Arduino UNO, Arduino Diecimila, Arduino Duemilanova, Arduino Leonardo, Arduino Mega dan Arduino Nano. Walaupun ada berbagai jenis kartu Arduino, secara prinsip pemrograman yang diperlukan menyerupai. Hal yang membedakan adalah kelengkapan fasilitas dan pin-pin yang perlu digunakan (Kadir, 2015:2).

Gambar 2. Arduino UNO

Bentuk *board* Arduino seperti gambar 2. Ardiono UNO berukuran sebesar kartu kredit. Walaupun berukuran kecil seperti itu, papan tersebut mengandung *microcontroller* dan sejumlah *input/output* (I/O) yang memudahkan pemakai untuk menciptakan berbagai proyek elektronika yang dikhususkan untuk menangani tujuan tertentu. Bagian-bagian dari Arduino UNO yang perlu diketahui seperti berikut:

a. Port USB digunakan untuk menghubungkan Arduino UNO dengan komputer, melalui sepasang kabel USB
b. Colokan catu daya *eksternal* digunakan untuk memasok sumber daya listrik untuk Arduino UNO ketika tidak dihubungkan ke komputer. Jika Arduino UNO dihubungkan ke komputer melalui kabel USB, pasokan daya listrik diberi oleh komputer.
c. Pin digital mempunyai label 0 sampai 13, disebut pin digital karena mempunyai isyarat digital, yakni berupa 0 atau 1. Dalam praktik, nilai

0 dinyatakan dengan tegangan 0V dan nilai 1 dinyatakan dengan tegangan 5V.

d. Pin analog berarti bahwa pin-pin ini mempunyai nilai yang bersifat analog (nilai yang berkesinambungan). Dalam program, nilai setiap pin analog yang berlaku sebagai masukan (hasil dari sensor) berkisar antara 0 sampai dengan 1023.

e. *Microcontroller* yang digunakan di Arduino UNO adalah ATMega328.

f. Ada dua pin yang dapat digunakan untuk memasok catu daya ke komponen elektronis yang digunakan dalam menangani proyek, misalnya sensor gas, sensor jarak dan relay. Tegangan yang tersedia adalah 3,3V dan 5V. Komponen-komponen elektronis yang diberi tegangan oleh Arduino UNO adalah yang memerlukan arus kecil. Sebagai contoh, motor DC yang menarik arus lebih dari 500mA harus menggunakan catu daya tersendiri.

g. Arduino UNO dilengkapi dengan *static random acces memory* (SRAM) berukuran 2KB untuk memegang data, *Flash memory* berukuran 32KB dan *erasable programmable read only memory* (EEPROM). SRAM digunakan untuk menampung data atau hasil pemrosesan data selama Arduino menerima pasokan catu daya. *Flash memory* untuk menaruh program yang anda buat. EEPROM

digunakan untuk menaruh program bawaan dari Arduino UNO dan sebagian lagi dapat dimanfaatkan untuk menaruh data milik anda secara permanen.

6. Sensor Ultrasonik HC-SR04

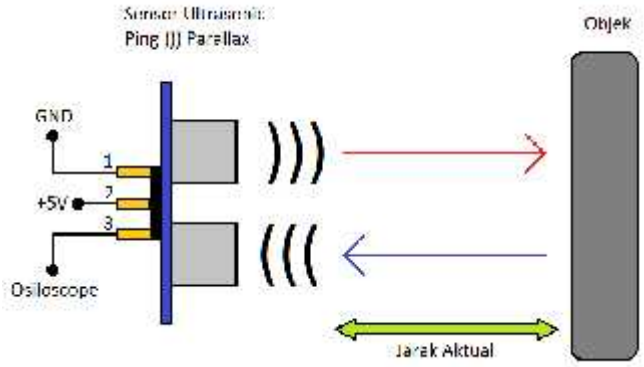
Sensor ultrasonik adalah sensor yang mengirimkan gelombang suara dan kemudian memantau pantulannya sehingga dapat digunakan untuk mengetahui jarak antara sensor dengan objek yang memantulkan kembali gelombang suara tersebut. Sensor ini bisa dipakai di berbagai aplikasi seperti pada mobil untuk menghindari tabrakan, untuk membunyikan alarm kalau ada orang mendekati pintu, dan mengukur tinggi orang (Kadir, 2015:100).

Gambar 3. Sensor Ultrasonik HC-SR04

Gambar 3 merupakan bentuk fisik dari sensor ultrasonik HC-SR04 yang mempunyai 4 pin, satu pin VCC sebagai pin masukan tegangan dan di imbangi pin GND untuk *grounding*, sedangkan dua pin sisanya adalah *trigger* dan *echo* pin yang akaan mempengaruhi gelombang ultrasonik.

Gelombang ultrasonik adalah gelombang bunyi yang mempunyai frekuensi sangat tinggi yaitu 20.000 Hz. Bunyi ultrasonik tidak dapat di

dengar oleh telinga manusia. Bunyi ultrasonik dapat didengar oleh anjing, kucing, kelelawar, dan lumba-lumba. Bunyi ultrasonik bisa merambat melalui zat padat, cair dan gas. Reflektivitas bunyi ultrasonik di permukaan zat padat hampir sama dengan reflektivitas bunyi ultrasonik di permukaan zat cair. Akan tetapi, gelombang bunyi ultrasonik akan diserap oleh tekstil dan busa.



Gambar 4. Prinsip kerja sensor ultrasonik

Prinsip kerja dari sensor ultrasonik sinyal yang akan dibangkitkan di pancarkan dari *transmitter* ketika sinyal mengenai benda penghalang, maka sinyal akan di pantulkan dan diterima oleh *receiver* dan di proses oleh rangkaian *microcontroller*.

7. Bluetooth HC-05

Bluetooth adalah protokol komunikasi *wireless* yang bekerja pada frekuensi radio 2.4 GHz untuk pertukaran data pada perangkat bergerak seperti PDA, leptop, HP, dan lain-lain. Salah satu hasil contoh modul

bluetooth yang paling banyak digunakan adalah tipe HC-05. Modul bluetooth HC-05 merupakan salah satu modul bluetooth yang dapat ditemukan dipasaran dengan harga yang relatif murah. Modul bluetooth HC-05 terdiri dari 6 pin konektor, yang setiap pin konektor memiliki fungsi yang berbeda-beda. Gambar 5 adalah bentuk dari modul bluetooth HC-05.

Gambar 5. Bluetooth HC-05

Modul bluetooth HC-05 merupakan modul bluetooth yang bisa menjadi *slave* ataupun *master* hal ini dibuktikan dengan bisa memberikan notifikasi untuk melakukan pairing ke perangkat lain, maupun perangkat lain tersebut melakukan pairing ke modul bluetooth HC-05. Untuk mengeset perangkat bluetooth dibutuhkan perintah-perintah AT *command* yang mana perintah AT *command* tersebut akan di respon oleh perangkat bluetooth jika modul bluetooth tidak dalam keadaan terkoneksi dengan perangkat lain (Kadir ,2015:393).

8. Motor Servo

Motor servo adalah sebuah motor DC dengan sistem umpan balik tertutup di mana posisi rotor-nya akan di di informasikan kembali ke rangkaian kontrol yang ada di dalam motor servo. Motor ini terdiri dari sebuah motor DC, serangkaian *gear*, potensiometer, dan rangkaian kontrol. Potensiometer berfungsi untuk menentukan batas sudut dari putaran sevo, sedangkan sudut dari sumbu motor servo diatur berdasarkan lebar pulsa yang dikirim melalui kaki sinyal dari kabel motor servo (Dickson, 2015:52). Gambar 6 adalah bentuk fisik dari motor servo.

Gambar 6. Motor Servo SG90 9g

Motor servo pada dasarnya dibuat menggunakan DC yang dilengkapi dengan kontroler dan sensor posisi sehingga dapat memiliki gerakan 0 derajat, 90 derajat, 180 derajat atau 360 derajat. Tiap komponen pada motor servo masing-masing memiliki fungsi sebagai kontroler, *driver*, sensor, girbox dan *aktuator*. Motor pada sebuah motor servo adalah motor DC yang dikendalikan oleh bagian kontroler, kemudian komponen yang berfungsi sebagai sensor adalah potensiometer yang terhubung pada sistem girbox pada motor servo.

Untuk menjalankan atau mengendalikan motor servo berbeda dengan motor DC. Karena untuk mengendalikan motor servo perlu diberikan

sumber tegangan dan sinyal kontrol. Besarnya sumber tegangan tergantung dari spesifikasi motor servo yang digunakan. Sedangkan untuk mengendalikan putaran motor servo dilakukan dengan mengirim pulsa kontrol dengan frekuensi 50 Hz dengan periode 20ms dan *duty cycle* yang berbeda, dimana untuk menggerakan motor servo sebesar 90 derajat diperlukan pulsa dengan to *duty cycle* pulsa positif 1,5ms dan untuk bergerak sebesar 180 derajat diperlukan lebar pulsa 2ms.

9. LED

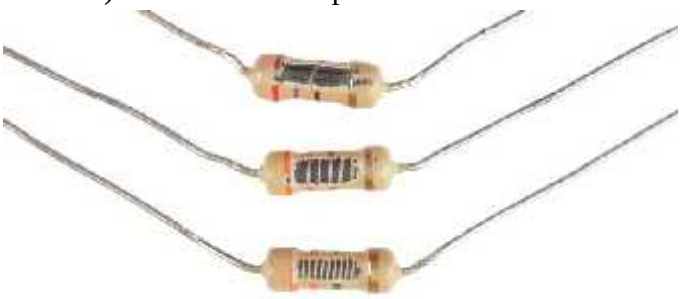
Light Eminiting Diode atau sering disingkat LED adalah komponen elektronika yang dapat memancarkan cahaya monokromatik ketika diberikan tegangan maju. LED merupakan keluarga dioda yang terbuat dari bahan semikonduktor. Warna-warna cahaya yang di pancarkan oleh LED tergantung pada jenis bahan semikonduktor yang digunakan. LED juga dapat memancarkan sinar inframerah yang tidak tampak oleh mata seperti yang sering kita jumpai pada *remote control* TV ataupun *remote control* perangkat elektronik lain (Kadir, 2015:51).

Gambar 7. LED

Gambar 7 merupakan bentuk fisik LED, bentuk LED mirip dengan sebuah bohlam (bola lampu) yang kecil dan dapat dipasangkan dengan mudah ke dalam berbagai perangkat elektronik. Bedanya dengan lampu pijar LED tidak memerlukan pembakaran filamen sehingga tidak menimbulkan panas dalam menghasilkan cahaya. Oleh karena itu, saat ini LED yang bentuknya kecil telah banyak digunakan sebagai lampu penerang dalam LCD TV yang menggantikan lampu *tube*.

10. Resistor

Resistor merupakan salah satu komponen yang paling sering ditemukan dalam rangkaian elektronika. Hampir setiap peralatan elektronika menggunakannya, pada dasarnya resistor adalah komponen elektronika pasif yang memiliki nilai resistensi atau hambatan yang berfungsi untuk membatasi dan mengatur arus listrik dalam suatu rangkaian elektronika. Resistor atau dalam bahasa Indonesia sering disebut dengan hambatan atau tahanan dan biasa disingkat dengan huruf “R” (Kadir, 2015:52). Gambar 8 merupakan bentuk resistor.

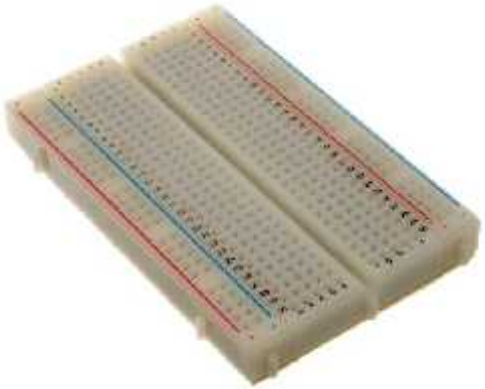


Gambar 8. Resistor

11. Breadboard

Sebuah *breadboard* sebenarnya mengacu pada sebuah papan tempat membuat rangkaian elektronika yang tidak perlu dilakukan penyolderan untuk menghubungkan antara satu komponen dengan

komponen lain sehingga dikenal dengan *solderless breadboard*. *Breadboard* adalah sebuah perangkat yang digunakan untuk membuat rangkaian sementara atau rangkaian *prototype* yang sama sekali tidak membutuhkan proses penyolderan, cara seperti ini akan menghemat waktu dan biaya (Hendriono, 2014:2). Gambar 9 adalah bentuk dari *breadboard*.



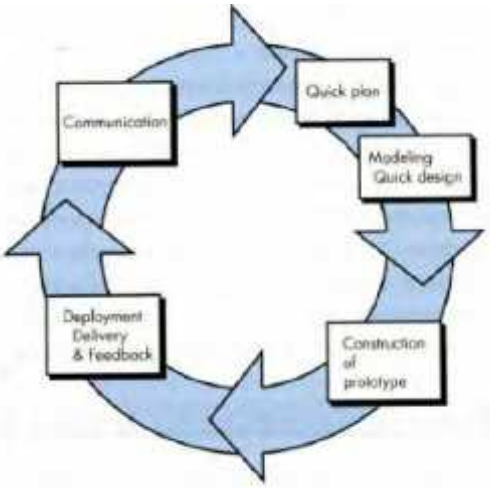
Gambar 9. *Breadboard*

Breadboad adalah papan khusus yang dgunakan untuk membuat prototipe atau rangkaian elektronik yag bersifat percobaan. *Breadboard* merupakan tempat atau media yang sangat mudah digunakan baik untuk rangkaian sederhana maupun rangkaian yang sangat kompleks, penggunaan umum lain dari *breadboard* adalah untuk menguji bagian-bagian atau fungsi-fungsi dari sirkuit terpadu atau sering disebut IC (*Interated Circuit*).

12. Model *Prototype*

Sering kali pelanggan mendefinisikan satu set tujuan umum untuk perangkat lunak, tetapi tidak mengidentifikasi persyaratan rinci untuk fungsi dan fitur. Di lain kasus, pengembang mungkin tidak yakin dari efisiensi sebuah algoritma, adaptasi dari sistem operasi, atau bentuk yang

interaksi antara manusia dan mesin. Dalam hal ini paradigma *protoype* menawarkan pendekatan yang terbaik (Prassman, 2010:43).



Gambar 10. Model *Prototype*

Tahapan-tahapan dalam *prototype* adalah sebagai berikut :

a) Komunikasi dan pengumpulan data awal, yaitu analisis kebutuhan sistem yang utuh ke dalam bagian - bagian komponen dengan maksud untuk mengidentifikasi permasalahan atau hambatan yang terjadi.
b) *Quick plan*, yaitu pembuatan desain *prototype* sesuai dengan kebutuhan sistem yang telah didefinisikan sebelumnya untuk selanjutnya dikembangkan kembali.
c) Pembentukan *prototype*, yaitu pembuatan perangkat *prototype* termasuk pengujian dan penyempurnaan.
d) Evaluasi terhadap *prototype*, yaitu mengevaluasi *prototype* dan memperhalus analisis terhadap kebutuhan pengguna. Evaluasi ini

dilakukan agar dapat mengetahui bahwa *prototype* yang telah dirancang atau dikembangkan sudah sesuai kebutuhan.

e) Perbaikan *prototype*, yaitu pembentukan tipe yang sebenarnya berdasarkan hasil evaluasi *prototype*.

f) Produksi akhir, yaitu memproduksi perangkat secara benar sehingga dapat digunakan oleh pengguna. (Pressman. 2010:43).

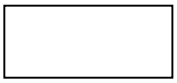
13. Bagan Alir Sistem atau Dokumen (*Flowchart*)

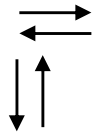
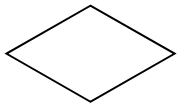
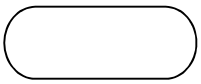
Bagan alir sistem atau *flowchart* merupakan bagan yang menunjukkan arus pekerjaan secara keseluruhan dari sistem dan menjelaskan urut-urutan dari prosedur-prosedur yang ada di dalam sistem dan menunjukkan apa yang dikerjakan di sistem. Bagan alir dokumen disebut juga bagan alir formulir merupakan bagan alir yang menunjukkan arus dari laporan dan formulir termasuk tembusan-tembusannya. (Jogiyanto. 2001:796).

Bagan alir program merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dari derisikasi bagan alir sistem (Jogiyanto. 2001:802).

Bagan alir program dibuat dengan menggunakan simbol-simbol sebagai tabel 1.

Tabel 1. Tabel Simbol-Simbol *Flowchart*

Input/output	Simbol input / output digunakan untuk mewakili data Input / input.
	Simbol proses digunakan untuk mewakili suatu proses.

	Simbol garis alir digunakan untuk menunjukkan arus dari proses.
	Simbol keputusan digunakan untuk suatu penyeleksian kondisi di dalam program.
	Simbol ini digunakan untuk menjalankan awal dan akhir dari suatu proses.

14. *Black-Box Testing*

Black-box testing berfokus pada persyaratan fungsional perangkat lunak yang memungkinkan engineers untuk memperoleh set kondisi input yang sepenuhnya akan melaksanakan persyaratan fungsional untuk sebuah program (Pressman 2010:495).

Black-box testing berusaha untuk menemukan kesalahan dalam kategori berikut:

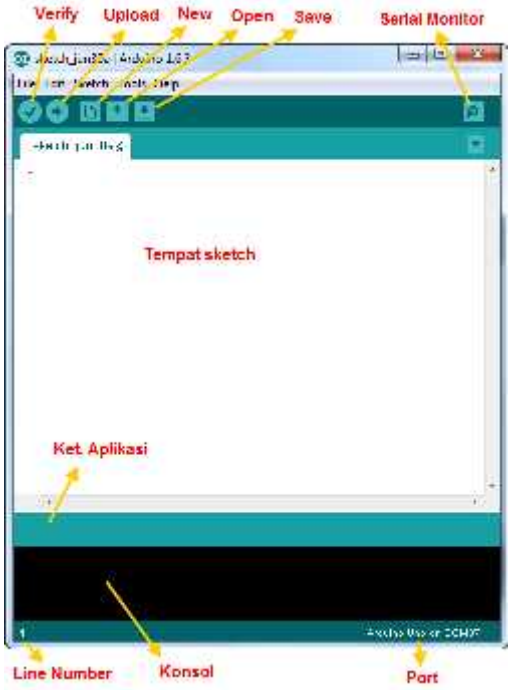
1. Fungsi yang tidak benar atau fungsi yang hilang
2. Kesalahan antarmuka
3. Kesalahan dalam struktur data atau akses database eksternal
4. Kesalahan perilaku (*behavior*) atau kesalahan kinerja
5. Inisialisasi dan pemutusan kesalahan

C. Perangkat Lunak yang Digunakan

Dalam pembuatan rancang bangun pintu garasi otomatis menggunakan Arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 ini penulis menggunakan perangkat lunak sebagai berikut :

1. Arduino IDE

Untuk memprogram *board* Arduino, kita butuh aplikasi IDE (*Integrated Development Environment*) bawaan dari Arduino. Aplikasi ini berguna untuk membuat, membuka, dan mengedit *source code* Arduino (*Sketches*, para *programmer* menyebut *source code* arduino dengan istilah "*sketches*"). Selanjutnya, jika kita menyebut *source code* yang ditulis untuk Arduino, kita sebut "*sketch*" juga. *Sketch* merupakan *source code* yang berisi logika dan algoritma yang akan diupload ke dalam IC *microcontroller* (Santoso, 2015:7).



Gambar 11. Arduino IDE

Interface Arduino IDE tampak seperti gambar 4. dari kiri ke kanan dan atas ke bawah, bagian-bagian IDE Arduino terdiri dari:

1) Verify : pada versi sebelumnya dikenal dengan istilah *Compile*. Sebelum aplikasi diupload ke *board* Arduino, biasakan untuk memverifikasi terlebih dahulu *sketch* yang dibuat. Jika ada kesalahan pada *sketch*, nanti akan muncul error. Proses Verify / *Compile* mengubah *sketch* ke *binary code* untuk diupload ke *microcontroller*.
2) Upload : tombol ini berfungsi untuk mengupload *sketch* ke *board* Arduino. Walaupun kita tidak mengklik tombol *verify*, maka *sketch* akan di-*compile*, kemudian langsung diupload ke *board*. Berbeda dengan tombol *verify* yang hanya berfungsi untuk memverifikasi *source code* saja.
3) New Sketch : Membuka window dan membuat *sketch* baru
4) Open Sketch : Membuka *sketch* yang sudah pernah dibuat. *Sketch* yang dibuat dengan IDE Arduino akan disimpan dengan ekstensi file .ino
5) Save Sketch : menyimpan *sketch*, tapi tidak disertai mengcompile.
6) Serial Monitor : Membuka *interface* untuk komunikasi serial, nanti akan kita diskusikan lebih lanjut pada bagian selanjutnya
7) Keterangan Aplikasi : pesan-pesan yang dilakukan aplikasi akan muncul di sini, misal "*Compiling*" dan "*Done Uploading*" ketika kita mengcompile dan mengupload *sketch* ke *board* Arduino
8) Konsol : Pesan-pesan yang dikerjakan aplikasi dan pesan-pesan tentang *sketch* akan muncul pada bagian ini. Misal, ketika aplikasi mengcompile atau ketika ada kesalahan pada *sketch* yang kita buat, maka informasi *error* dan baris akan diinformasikan di bagian ini.

9) Baris Sketch : bagian ini akan menunjukkan posisi baris kursor yang sedang aktif pada *sketch*.
10) Informasi Port : bagian ini menginformasikan *port* yang dipakah oleh *board* Arduino.

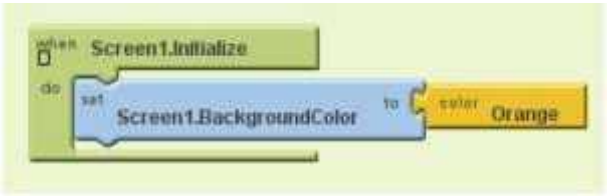
Struktur perintah pada Arduino secara garis besar terdiri dari 2 bagian yaitu *void setup* dan *void loop*. *Void setup* berisi perintah yang akan di eksekusi hanya satu kali sejak Arduino di hidupkan sedangkan *void loop* berisi perintah yang akan di eksekusi berulang-ulang selama Arduino dinyalakan.

2. MIT App Inventor 2 beta

App Inventor adalah program yang dibuat oleh Google dan sekarang dikembangkan oleh MIT. Program ini dapat digunakan untuk membuat dan mendesain aplikasi Andriod yang berbasis Web page dan Java interface. Hanya dengan pengetahuan pemograman yang sedikit pengguna sudah bisa membuat sebuah aplikasi Android yang sederhana. Jika pengguna sudah berpengalaman menggunakan App Inventor pengguna juga bisa membuat program yang sangat rumit dan berguna hanya dengan menggukan App Inventor (Prasetiyo, 2011:1).

Yang menarik dari App Inventor adalah, pengguna tidak lagi menemukan kode-kode program seperti halnya ketika menggunakan *tools* developing lain. Kode-kode itu sudah dibungkus kedalam objek visual, yang dinamakan *Blocks*. Pengguna tinggal memahami fungsi dari *Blocks* tersebut, kemudian bagaimana dia berinteraksi dengan *Blocks* lain, dan terakhir merangkainya persis seperti merangkai puzzle untuk membuat suatu bentuk atau gambar. Coba lihat pada gambar, seperti potongan

puzzle bukan? sangat mudah dan menyenangkan, disini bisa kita anggap pengguna melewati fase untuk belajar mengenai bahasa pemrograman, dan berfokus pada belajar kreatifitas dan logika.



Gambar 12. Contoh *Blocks*

Berikut adalah 3 bagian komponen utama dalam MIT App Inventor :

1) Desainer : Membuat UI (*User Interface*) dan memasukkan komponen-komponen App Inventor yang akan pengguna pakai pada aplikasi dengan jendela Desainer pengguna. Disini terdapat bagian *Palette, Viewer, Components dan Properties*.

Gambar 13. Jendela Desainer

Selain itu juga terdapat beberapa menu penting yang akan pengguna gunakan.

a) *Pallete* : Berisi seluruh komponen yang bisa pengguna pakai untuk membuat aplikasi.
b) *Viewer* : Untuk menempatkan Komponen dan mendesain seperti apa tampilan atau UI dari aplikasi.
c) *Components* : Berisi komponen yang telah pengguna ambil dan akan gunakan pada aplikasi.
d) *Properties* : Disini pengguna bisa mengubah properti dari masing-masing komponen, misal warna layar.
e) *Add/Remove Screen* : Menambahkan atau menghapus *screen*, ini fitur baru pada App Inventor, yaitu dukungan *Multiscreen*.
f) *Open The Blocks Editor* : untuk mengaktifkan jendela *Blocks Editor*, sekali diaktifkan apabila *Blocks Editor*.
g) *Package* for Phone : Ketika aplikasi selesai dibuat, dan pengguna ingin mencobanya di handset android, maka bisa menggunakan menu ini.

2) *Blocks Editor* : adalah tempat untuk mengambil *blocks* (*kode program yg telah dijadikan objek visual*), dan menyusun *Blocks-Blocks* tersebut sehingga berfungsi sesuai dengan yang pengguna inginkan.

Gambar 14. *Blocks Editor*

Ada lima menu penting disini yaitu :

a) *Built-In* : *Blocks* dasar dari komponen App Inventor

b) *My Blocks* : *Blocks* dari komponen yang pengguna gunakan untuk aplikasi, yaitu yang dimasukkan pada Desainer

c) *Advance* : *Blocks* pelengkap dari yang ada pada *My Blocks*

d) *New Emulator* : Menu untuk mengaktifkan *Emulator*

e) *Connect To Devices* : Menu untuk mengkoneksikan projek aplikasi dengan *Emulator* atau HP Android, yaitu ketika pengguna ingin mencoba Aplikasi yang sedang dibangun. Dan setelah sekali dikoneksikan maka ketika terjadi perubahan pada aplikasi yang sedang dibangun, misal komponen diganti, *blocks* diganti, secara otomatis akan dikomunikasikan ke *Emulator* atau HP Android.

3) *Emulator Handset* Android : Apabila pengguna ingin mencoba aplikasi android yang telah dibuat, pengguna bisa memakai *Emulator*, hampir semua yang bisa dilakukan oleh Handset Android bisa dilakukan oleh Emulator, kecuali menelepon, berkirim SMS, dan GPS. Jadi pengguna tidak usah kuatir, ketika membuat aplikasi Android dengan App Inventor, maka pengguna bisa menggunakan Emulator sebelum mencobanya di *Handset* atau HP Android.

Gambar 15. *Emulator*

Apabila pengguna sudah memahami *workspace* dari App Inventor ini, maka tinggal pengguna mencobanya. Ada banyak contoh dari aplikasi App Inventor ini di internet. Dengan melihat komponen

yang digunakan dan dipasang pada Desainer, atau melihat *Blocks* yang tersusun pada *Blocks Editor*, pengguna akan segera bisa membuat dan mencobanya sendiri. Jika ingin cepat menguasainya, kuncinya sekali lagi adalah berlatih dan berlatih, pikirkan kita akan membuat aplikasi apa, kemudian cobalah untuk berlatih membuatnya.

Bagi yang belum pernah mengenal App Inventor, App Inventor adalah satu *Tool webbased* (*Cloud*) untuk membuat Aplikasi pada *platform* Android yang ditujukan untuk semua orang. Pengguna bisa membuat aplikasi android tanpa kode sama sekali.

BAB III
ANALISA DAN PERANCANGAN SISTEM

A. Tinjauan Umum

Kebutuhan hidup yang semakin meningkat membuat manusia semakin sibuk sehingga terkadang membuka dan menutup pintu garasi mobil secara berulang-ulang menjadi hal yang cukup merepotkan baik dari segi tenaga maupun waktu, apalagi jika pemilik rumah yang menggunakan mobil tidak memiliki pembantu maka pemilik rumah harus turun dari mobil untuk membuka dan menutup pintu garasi secara manual. Jika saja pemilik rumah dapat membuka dan menutup pintu garasi tanpa harus turun dari mobil, maka akan lebih efisien baik dari segi waktu maupun tenaga.

B. Analisis

Pada tahap ini, penulis akan melakukan analisa terhadap sistem yang akan di bangun nanti. Adapun analisa tersebut adalah sebagai berikut :

1. Analisa Kelemahan Sistem

Pada umumnya, untuk membuka dan menutup pintu garasi dilakukan secara manual yaitu dengan menarik atau mengangkat pintu garasi tersebut, namun hal ini dirasa kurang efisien baik dari segi waktu maupun tenaga.

Dari penjelasan tersebut, maka dibuat rancang bangun pintu garasi otomatis yang dapat dikendalikan melalui *smartphone* tanpa harus membuka dan menutup pintu garasi secara manual.

2. Analisis Kebutuhan Sistem

a. Kebutuhan Perangkat Keras

Kebutuhan perangkat keras yang digunakan dalam pengembangan sistem ini terdiri dari 2 bagian yaitu kebutuhan perangkat keras untuk mengembangkan sistem dan kebutuhan perangkat keras untuk implementasi sistem. Berikut uraian dari kebutuhan tersebut :

1) Kebutuhan perangkat keras untuk pengembangan sistem
   a) Laptop Acer Aspire V5-471PG
   b) Processor Intel® Core™ i5-3317U CPU @ 1.70GHz
   c) RAM DDR3 4.00 GB
   d) Kabel USB A-B
2) Kebutuhan perangkat keras untuk implementasi sistem

   Perangkat keras yang dibutuhkan untuk pengimplementasian pintu garasi otomatis ini adalah *smartphone* dengan sistem operasi Android dari versi 4.1 sampai dengan Android versi terbaru. Adapun beberapa komponen lainnya sebagai berikut :

   a) Sensor Ultrasonik HC-SR04 2 buah
   b) Bluetooth modul HC-05
   c) Motor servo
   d) *Board* Arduino UNO R3
   e) LED 2 buah

f) *Prototype* Garasi

g) Mobil RC

b. Kebutuhan Perangkat Lunak

Adapun kebutuhan perangkat lunak yang digunakan dalam pengembangan sistem adalah sebagai berikut :

1) Arduino IDE
2) MIT App Inventor 2 beta
3) Sistem operasi Windows 10 Home

c. Kebutuhan Informasi

Sistem yang dibangun akan menampilkan informasi berupa status pintu garasi dalam keadaan terbuka atau tertutup, dalam kasus ini penulis menggunakan *smartphone* Android untuk membuka dan menutup pintu garasi.

d. Kebutuhan pengguna

Sistem yang dibangun ini dapat digunakan oleh pemilik rumah yang membutuhkan, khususnya bagi pemilik rumah yang memiliki banyak pekerjaan dan tidak memiliki pembantu. Untuk dapat menggunakan pintu garasi otomatis ini, kebutuhan yang diperlukan oleh pengguna adalah sebuah *board* Arduino UNO R3, sensor ultrasonik HC-SR04 2 buah, bluetooth modul HC-05, dan motor servo yang sudah menjadi sebuah pintu garasi otomatis.

3. Analisis Kelayakan Sistem

Pada sub bab ini akan membahas analisis tentang kelayakan teknologi, kelayakan hukum dan kelayakan operasional.

a. Kelayakan Teknologi

Dari kelayakan pintu garasi otomatis ini dapat dikatakan layak karena untuk menggunakan pintu garasi otomatis ini tidak diperlukan *smartphone* Android yang berspesifikasi tinggi, dan untuk mendapatkan *smartphone* Android dengan spesifikasi yang dibutuhkan sangat mudah karena harga cukup murah dan terjangkau.

b. Kelayakan Hukum

Dari segi kelayakan hukum pintu garasi otomatis ini dapat dikatakan layak karena merupakan hasil karya penulisan sendiri dan tidak merupakan pelanggarak hak cipta orang lain.

c. Kelayakan Operasional

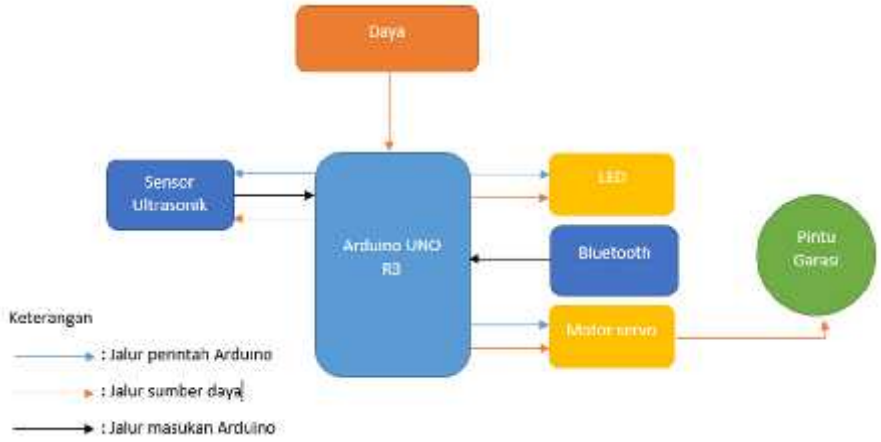
Dari segi operasional kelayakan pintu garasi otomatis ini dapat dikatakan layak karena saat ini sebagian besar masyarakat sudah mampu mengoperasikan *smarthphone* Android dengan baik, selain dapat menggantikan pengoperasian secara manual, juga lebih efisien baik dari segi waktu dan tenaga.

C. Desain Sistem

Pada tahap ini, penulis akan melakukan desain terhadap sistem yang akan di bangun nanti. Adapun desain dari sistem sebagai berikut :

1. Desain Proses

a. Diagram Blok



Gambar 16. Diagram Blok

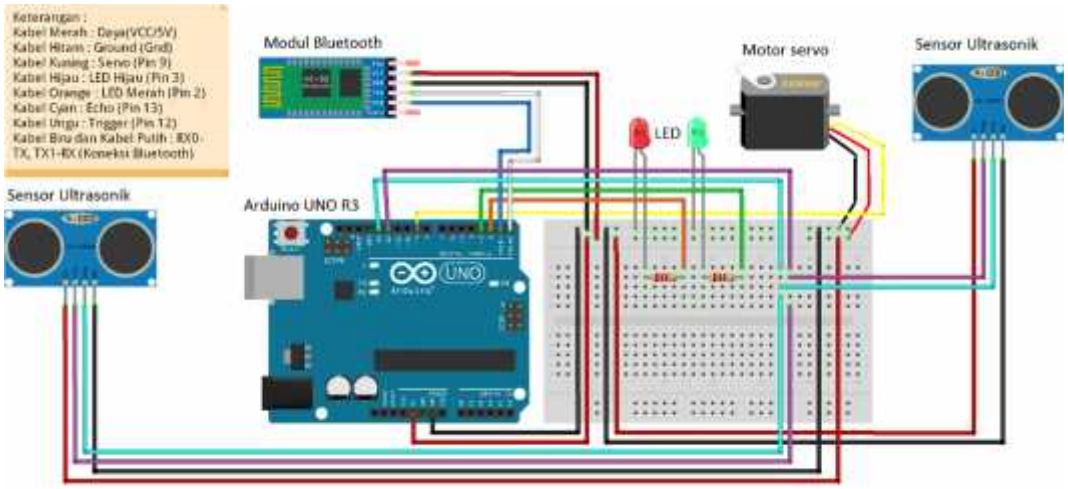
Dari gambar 16. diagram blok rancang bangun pintu garasi otomatis menggunakan Arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 dapat dijabarkan prinsip kerja diagram blok tersebut sebagai berikut :

1) Sensor ultrasonik akan mendeteksi objek yang akan mengenainya dimana sensor akan diletakan di depan pintu garasi bagian luar dan di dalam. Sensor ultrasonik yang digunakan adalah HC-SR04 sebanyak 2 buah, apabila ada objek mengenai sensor maka akan menjadi masukan untuk *board* Arduino UNO R3.

2) *Board* Arduino UNO R3 akan mengolah atau memproses masukan dari sensor ultrasonik HC-SR04, kemudian akan meneruskan masukan ke LED.
3) LED akan berkedip sebagai indikator ketika sensor ultrasonik mendeteksi adanya objek.
4) Bluetooth sebagai pengendali pintu garasi yang digerakan menggunakan motor servo.

b. Rangkaian Sistem

Berikut adalah rangkaian sistem dan bentuk dari komponen-komponen yang digunakan dalam membuat pintu garasi otomatis :



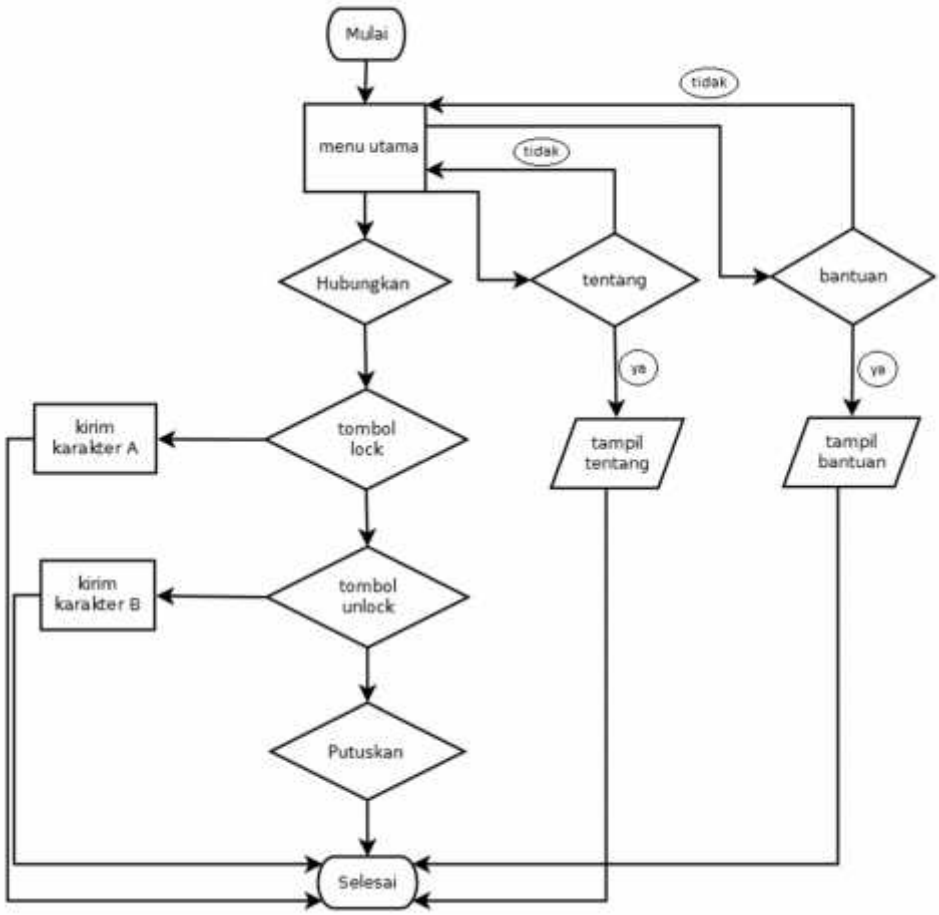
Gambar 17. Rangkaian Sistem

Dari gambar 17. rangkaian sistem rancang bangun pintu garasi otomatis menggunakan Arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 dapat dilihat terdapat 2 buah sensor ultrasonik HC-SR04 yang dihubungkan ke *breadboard* yaitu kabel

daya dan *groud* lalu kabel *trigger* dan echo di hubungkan ke pin 12 dan 13 pada *board* Arduino sebagai pendeteksi objek, 1 buah bluetooth HC-05 yang dihubungkan ke *breadboard* yaitu kabel daya dan groud lalu kabel RXD, TXD ke pin 1 dan 2 dihubungkan ke *board* Arduino digunakan untuk mengendalikan pintu garasi, 2 buah LED sebagai indikator objek dihubungkan ke *breadboard* dengan tambahan 2 buah resistor untuk menyamaratakan arus ke 2 bauh LED dan di hubungkan lagi ke *board* Arduino ke pin 3 dan 4, lalu 1 buah motor servo yang dihubungkan ke *breadboard* yaitu kabel daya dan ground dan 1 pin sisanya di hubungkan ke pin 9 pada *board* Arduino sebagai penggerak pintu garasi.

c. *Flowchart* Program Aplikasi

Agar pintu garasi dapat di kontrol melalui *smartphone* Android maka ditambahkan aplikasi di *smartphone* Android. Dengan memanfaatkan teknologi Android maka kita dapat membuat aplikasi pengendali pintu garasi otomatis dengan menggunakan MIT App Inventor 2 beta dan pengguna dapat membuka atau menutup pintu garasi melalui *smartphone*. Gambar 18 adalah *flowchart* dari program aplikasi :

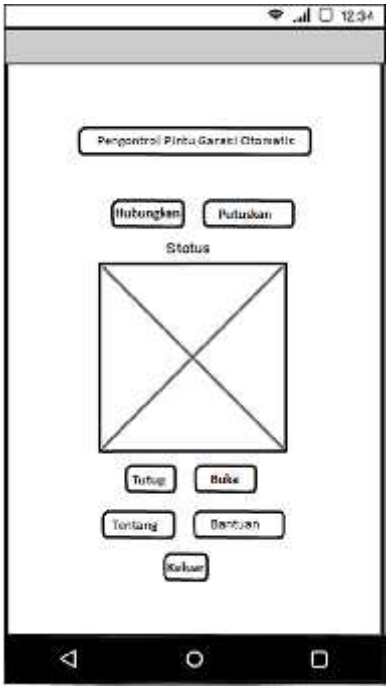


Gambar 18. *Flowchart* Program Aplikasi

Pada saat aplikasi mulai digunakan pengguna harus mengkoneksikan terlebih dahulu aplikasi dengan menekan tombol Hubungkan, setelah terhubung ke pintu garasi otomatis pengguna dapat membuka pintu garasi dengan menekan tombol *unlock* lalu aplikasi akan mengirimkan inputan berupa karakter B dan tombol *lock* digunakan untuk menutup pintu garasi lalu aplikasi akan mengirimkan inputan berupa karakter A, terdapat menu tambahan yaitu tentang dan bantuan yang dapat di akses melalui menu utama.

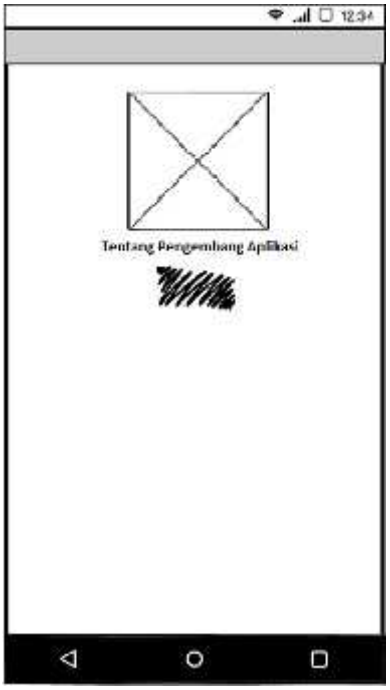
2. Desain *Interface* / Antarmuka

Pada sub bab ini penulis akan menguraikan tentang desain rancangan antarmuka program yang penulis rencanakan yaitu tampilan utama, tampilan tentang dan tampilan bantuan.



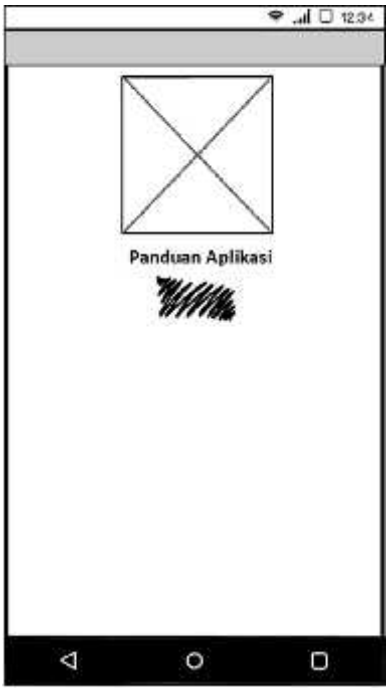
Gambar 19. Rancangan Tampilan Utama

Tampilan utama merupakan pengendali sistem pintu garasi otomatis. Pada tampilan utama ini terdapat dua buah tombol utama yaitu Tutup dan Buka, sedangkan tombol Hubungkan dan Putuskan digunakan untuk melakukan koneksi ke perangkat yang terhubung serta tombol Keluar untuk keluar dan tombol untuk mengakses tentang dan bantuan.



Gambar 20. Rancangan Tampilan Tentang

Tampilan tentang berisi informasi mengenai kontak *programmer* .



Gambar 21. Rancangan Tampilan Bantuan

Tampilan bantuan berisi petunjuk cara penggunaan aplikasi pengendali pintu garasi otomatis. Serta terdapat tahapan-tahapan bagaimana melakukan koneksi ke perangkat dan cara membuka serta menutup pintu garasi.

BAB IV
IMPLEMENTASI DAN PEMBAHASAN

A. Implementasi

1. Uji Coba Program

Menurut Prassman (2012:580), agar program dapat berjalan sesuai dengan yang diharapkan, maka perlu dilakukan uji coba terhadap program yang dibuat. Pengujian perangkat lunak merupakan presentasi terbesar dari upaya teknis dalam proses perangkat lunak dimana tujuan dari pengujian perangkat lunak adalah untuk menemukan kesalahan.

Adapun metode yang digunakan dalam melakukan uji coba program ini menggunakan metode pendekatan *black box testing*. Metode pendekatan *black box testing* merupakan metode pengujian perangkat lunak (*software*) yang berfokus pada persyaratan fungsional untuk mengerjakan serangkaian kondisi masukan yang akan mencoba semua persyaratan fungsional *software*. Setelah melakukan uji coba program secara keseluruhan maka hasil diperoleh bahwa program dapat berjalan sesuai dengan yang diharapkan. Adapun hasil dapat dilihat pada tabel 2.

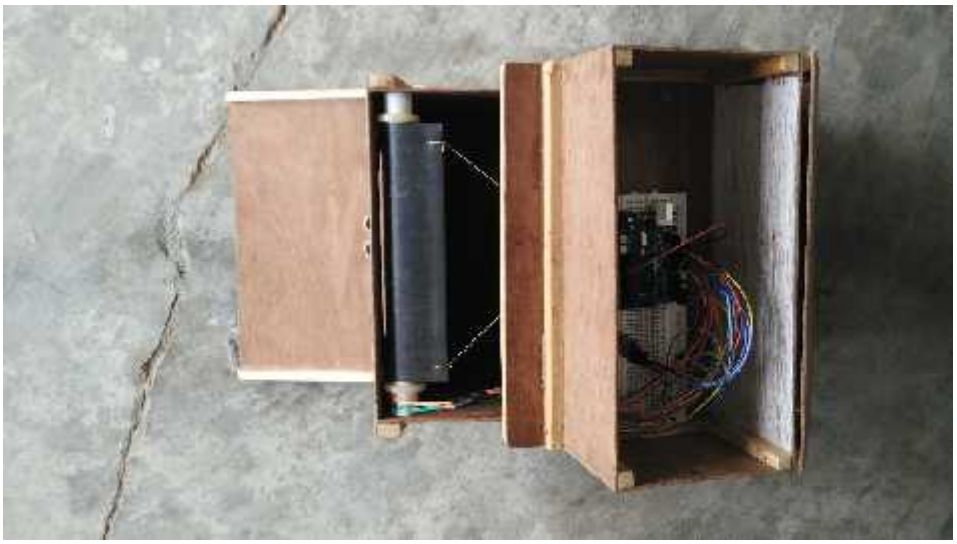
Tabel 2. Uji Coba Pintu Garasi Otomatis

No	Tes persyaratan fungsional	Nilai input	Hasil	Status
	1	**2**	**3**	**4**
1.	Tes koneksi ke board Arduino	Set port yang terdeteksi	Port terdeteksi	Sukses
2.	Tes koneksi ke Bluetooth	Hubungkan ke *smarhphone*	Perangkat terdeteksi	Sukses

	1	2	3	4
3.	Tes deteksi sensor 1	Ada Objek	LED berkedip	Sukses
4.	Tes deteksi sensor 2	Ada Objek	LED berkedip	Sukses
5.	Tes membuka pintu garasi	Tekan tombol *Unlock*	Pintu terbuka	Sukses
6.	Tes menutup pintu garasi	Tekan tombol *Lock*	Pintu tertutup	Sukses

2. Implementasi Sistem

Implementasi ini akan dilakukan dengan sebuah *prototype* garasi dan sebuah mobil RC. Gambar 22 adalah bentuk fisik dari *prototype* pintu garasi otomatis.



Gambar 22. Pintu Garasi Otomatis

Untuk mengetahui secara rinci cara kerja alat tersebut dapat langsung mengunjungi alamat berikut : *http://youtu.be/xlaRbxUejyk*

3. Panduan Program

Dalam menggunakan pintu garasi otomatis ini, pengguna harus mengerti terlebih dahulu cara mengoperasikan aplikasi *controller* pada Android.

Berikut langkah-langkah untuk mengoperasikan pintu garasi otomatis :

1) Nyalakan bluetooth pada *smartphone* Android terlebih dahulu.
2) Lakukan *pairing* bluetooth di *smartphone* Android dengan perangkat bluetooth yang ada di garasi untuk melakukan koneksi dengan memasukan password 1234 untuk kondisi *default*.
3) Tekan tombol *Connect* pada aplikasi *smartphone*, lalu pilih perangkat bluetooth yang sudah di *pairing*, jika koneksi berhasil tombol *Connect* akan berubah warna menjadi hijau.
4) Tekan tombol *Lock* untuk menutup pintu garasi dan *Unlock* untuk membuka pintu garasi.

4. Panduan Instalasi

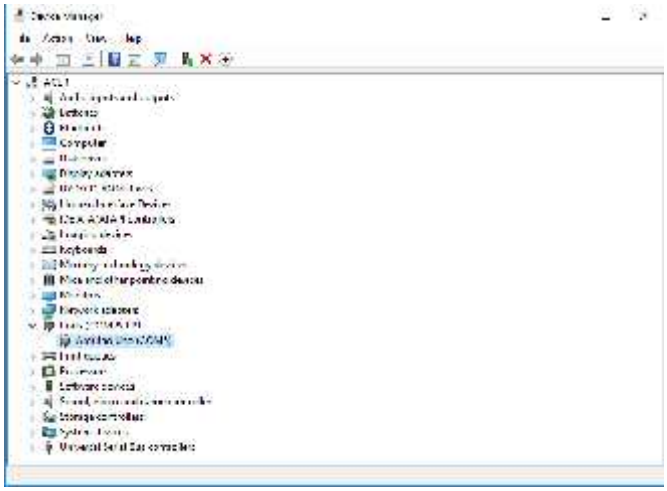
a. Panduan Instalasi Arduino

Agar sistem pintu garasi otomatis dapat berjalan, maka diperlukan sebuah *software* untuk membuat *source code* pada sistem yang berisi logika dan algoritma, untuk membuat *source code* tersebut penulis menggunakan Arduino IDE 1.6.8, adapun langkah-langkah yang penulis lakukan sebagai berikut :

1) Buka Arduino IDE 1.6.8.exe maka akan muncul jendela *sketch* seperti gambar berikut.

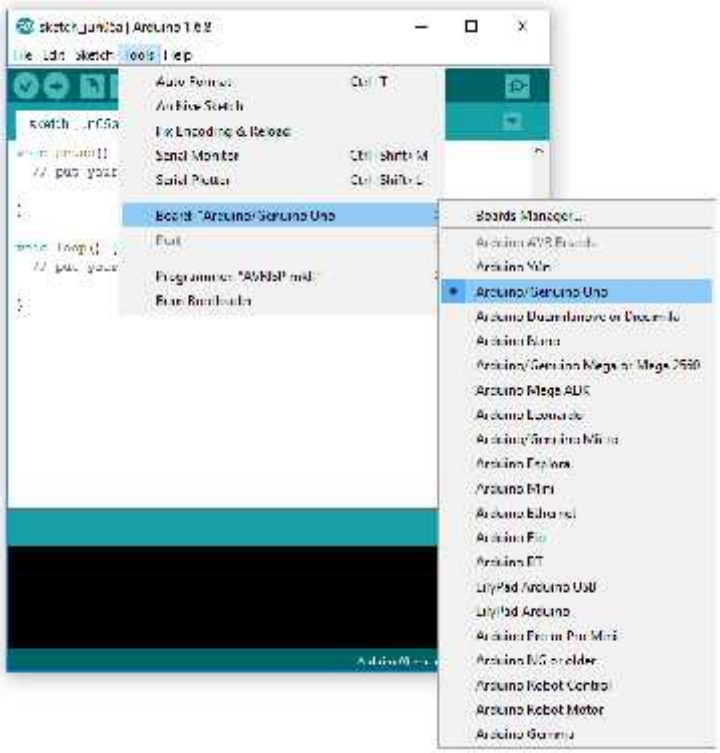
Gambar 23. *Sketch* Arduino IDE

2) Hubungkan *board* Arduino UNO dengan PC atau Leptop, kemudian buka Device Manager untuk mengetahui *port* pada *board* Arduino UNO seperti gambar berikut.



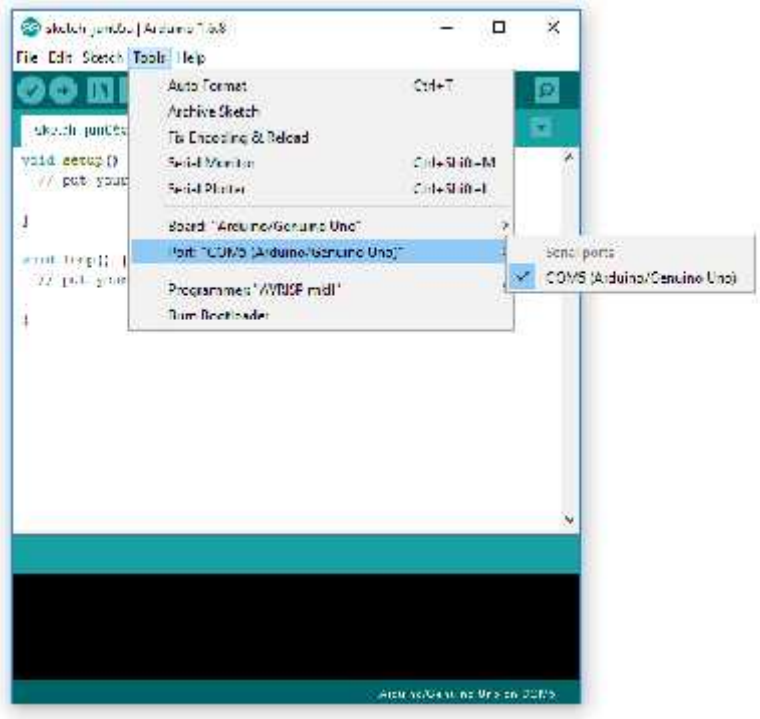
Gambar 24. *Port* Arduino UNO

3) Berikutnya setelah *port* Arduino UNO terdeteksi pilih jenis *board* yang akan digunakan pada Arduino IDE seperti gambar berikut.



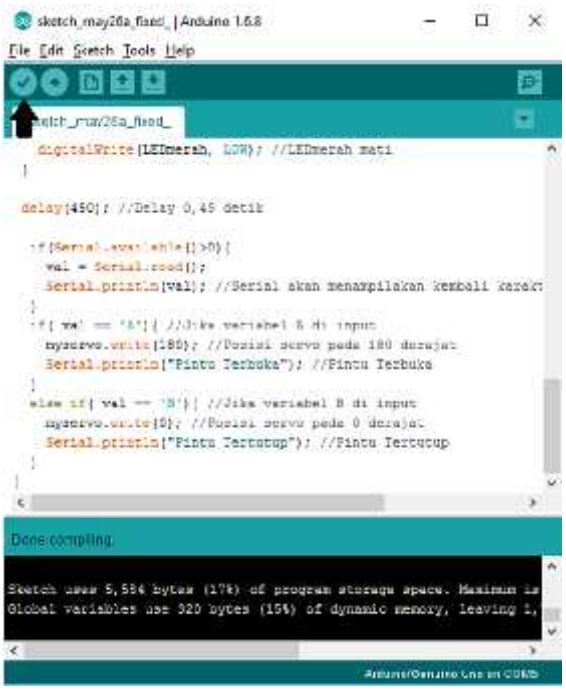
Gambar 25. *Setting Board* Arduino IDE

4) Setelah memilih *board* Arduino yang akan digunakan, pilih *port* untuk *board* Arduino seperti gambar berikut.



Gambar 26. *Setting Port* Arduino IDE

5) Selanjutnya adalah membuat *source code* atau biasa disebut *sketch* pada Arduino IDE, file yang disimpan Arduino IDE berekstensi .ino jika sudah tersimpan lakukan *verify/compile* seperti gambar berikut.



Gambar 27. Proses *compile* Arduino IDE

6) Jika tidak ada terjadi *error* pada *source code* atau *sketch* pada saat *compile*, *upload sketch* ke *board* Arduino UNO seperti gambar berikut.

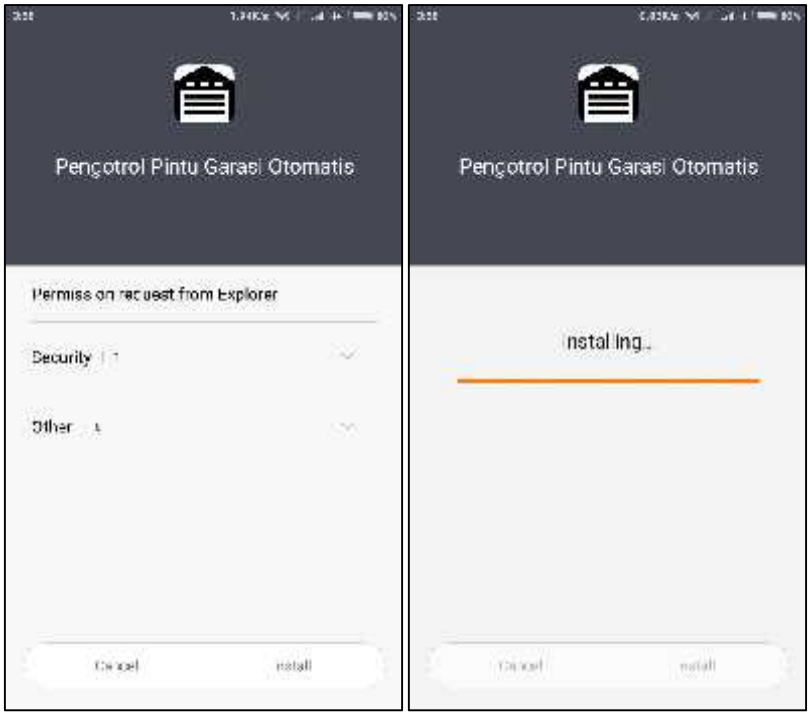
Gambar 28. Proses *upload sketch* Arduino IDE

b. Panduan Instalasi Aplikasi

Agar sistem pintu garasi otomatis dapat di kontrol melalui *smartphone* Android, maka diperlukan sebuah aplikasi *controller* untuk mengoperasikannya. Berikut cara instalasi aplikasi :

1) *Copy* file Bluetooth.apk ke *SD Card* atau *memory internal* pada *smartphone* Android.

2) *Tap* atau sentuh Bluetooth.apk, pilih *install* maka instalasi akan berjalan seperti gambar berikut.



Gambar 29. Proses Instalasi Aplikasi *Controller*

3) Tunggu sampai proses instalasi selesai seperti berikut.

Gambar 30. Proses Instalasi Selesai

B. Pembahasan

Dalam sebuah *source code* program terdapat logika dan algoritma, berikut penulis akan membahas *source code* program pada Arduino dan *blocks* program pada MIT App Inventor 2 beta yang digunakan dalam membangun sistem pintu garasi otomatis ini.

1) Pembahasan *Source code* Program pada Arduino

Pada sub bab ini penulis akan membahas *source code* yang ada pada *board* Arduino UNO yang dibuat menggunakan *software* Arduino IDE sebagai berikut.

```
#include <Servo.h> //Library untuk Servo
#include <NewPing.h> //Library untuk Sensor Ultrasonic
#define trigPin 12 //Set Trigger HCSR04 di Pin digital 12
#define echoPin 13 //Set Echo HCSR04 di Pin digital 13
#define MAX_DISTANCE 500 //Set jarak maksimal
NewPing sonar(trigPin, echoPin, MAX_DISTANCE);
char val; //Buat variabel untuk inputan bluetooth
int LEDhijau = 3, LEDmerah = 2; //Set LED hijau di pin 3 d
Servo myservo; //Buat object 1 buah motor servo
```

Baris kode #include <Servo.h> dan #include <NewPing.h> yaitu perintah yang digunakan untuk memanggil library motor servo dan sensor ultrasonik, baris kode #define trigPin 12 yaitu melakukan *setup* pada *board* Arduino bahwa pin 12 digunakan untuk *Trigger* sensor ultrasonik dan #define echoPin 13 yaitu *setup* pin 13 digunakan untuk *Echo* sensor ultrasonik, #define MAX_DISTANCE 500 yaitu *setup* untuk jarak maksimal yang akan di deteksi sensor ultrasonik, baris kode char val; adalah pendefinisian variabel val sebagai inputan untuk bluetooth sedangkan int LEDhijau = 3, LEDmerah = 2; yaitu pendefinisian variabel dan *setup* pin 3 digunakan untuk LED berwarna hijau dan pin 2 digunakan untuk LED berwarna merah, dan baris kode Servo myservo; yaitu pendefinisian variabel myservo untuk motor servo.

```
void setup() {
  pinMode(LEDmerah,OUTPUT); //Set LED merah sebagai output
  pinMode(LEDhijau,OUTPUT); //Set LED hijau sebagai output
  pinMode(trigPin, OUTPUT); //Set pin Trigger sebagai output
  pinMode(echoPin, INPUT); //Set pin Echo sebagai input
  Serial.begin(9600); //Kecepatan komunikasi serial
  myservo.attach(9); //Set servo di pin PWM 9
  myservo.write(0); //Set servo 0 derajat saat hidup
}
```

void setup( ) berisi perintah yang akan di eksekusi hanya satu kali sejak Arduino di hidupkan, baris kode pinMode (LEDmerah, OUTPUT);

, pinMode (LEDhijau, OUTPUT); , pinMode (trigPin, OUTPUT); , pinMode (echoPin, INPUT); mendefinisikan LEDmerah, LEDhijau dan trigPin sebagai *output*, sedangkan echoPin sebagai *input*, baris kode Serial.begin(9600); yaitu pengaturan untuk kecepatan *baudrate* pada modul bluetooth, baris kode myservo.attach(9); yaitu *setup* motor servo pada pin 9 sedangkan mysevo.write(0); agar posisi motor servo saat aktif dalam posisi 0 derajat.

```
void loop() {
  int duration, distance;
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = (duration/2) / 29.1; // Jarak dalam cm (duration/58)
  Serial.print(distance);
  Serial.println(" cm"); //Menampilak jarak secara realtime

  if(distance<=10){ //Jarak (cm) dapat disesuaikan
   digitalWrite(LEDhijau, LOW); //LEDhijau mati
   digitalWrite(LEDmerah, HIGH); //LEDmerah hidup
   Serial.println("Objek Terdeteksi!"); //Sensor mendeteksi objek
   delay(450); //Delay 0,45 detik
   digitalWrite(LEDmerah, LOW); //LEDmerah mati
 }
  else{ //Jika jarak lebih dari yang ditentukan
    digitalWrite(LEDhijau, HIGH); //LEDhijau hidup
    digitalWrite(LEDmerah, LOW); //LEDmerah mati
  }

  delay(450); //Delay 0,45 detik
```

void loop( ) berisi perintah yang akan di eksekusi berulang-ulang selama Arduino dinyalakan, baris kode int duration, distance; sampai Serial.prinln(" cm"); adalah *setup* untuk sensor ultrasonik, berikutnya penjelasan dari baris kode if (distance<=10) sampai delay (450); berisi

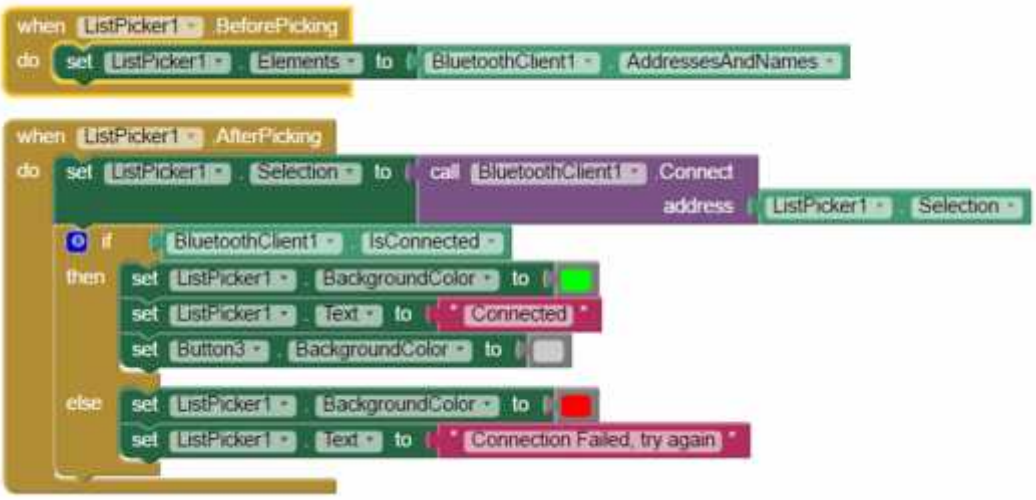
perintah dimana jika jarak objek kurang dari atau sama dengan 10 cm maka LEDhijau akan mati pada baris kode digitalWrite(LEDhijau, LOW); dan LEDmerah akan berkedip pada baris kode digitalWrite(LEDmerah, HIGH); digitalWrite(LEDmerah, LOW); dengan jeda 0,45 detik per 1 kali kedipan di berikan perintah delay (450); di antara LED merah saat hidup dan mati, jika tidak ada objek maka LEDhijau akan menyala dan LEDmerah akan mati pada baris kode setelah else.

```
  if(Serial.available()>0){
    val = Serial.read();
    Serial.println(val); //Serial akan menampilakan kembali
  }
  if( val == 'A'){ //Jika variabel A di input
    myservo.write(180); //Posisi servo pada 180 derajat
    Serial.println("Pintu Terbuka"); //Pintu Terbuka
  }
  else if( val == 'B'){ //Jika variabel B di input
    myservo.write(0); //Posisi servo pada 0 derajat
    Serial.println("Pintu Tertutup"); //Pintu Tertutup
  }
}
```

Berikutnya baris kode if(val == 'A'){ myservo.write(180); jika bluetooth diberi masukan “A” maka pintu garasi akan terbuka, sedangkan baris kode else if(val == 'B'){ myservo.write(0); jika diberi masukan “B” maka pintu garasi akan tertutup.

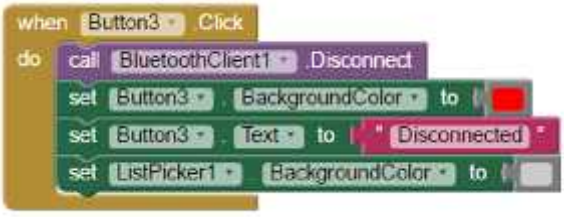
2) Pembahasan *Blocks* Program pada MIT App Inventor

Pada sub bab ini penulis akan membahas *blocks* program yang digunakan dalam membangun aplikasi *controller* pada MIT App Inventor sebagai berikut.



Gambar 31. *Blocks* Tombol *Connect*

Blocks pertama when "ListPicker1". BeforePicking do yaitu dimana sebelum "ListPicker1" / menghasilkan data "ListPicker1" akan mengambil data yaitu "AddressesAndNames" dari "BluetoothClient1", sedangkan *blocks* kedua when "ListPicker1". AfterPicking do yaitu setelah data yang berupa "AddressesAndNames" dari "BluetoothClient1" di dapat maka dipilih untuk digunakan di *blocks* "ListPicker1" Selection, jika bluetooth dalam keadaan terkoneksi maka "ListPicker1" / tombol *Connect* akan berubah warna menjadi hijau, dan jika gagal maka akan berubah warna menjadi merah dan text pada "ListPicker1" / tombol *Connect* akan berubah menjadi "Connection Failed, try again".

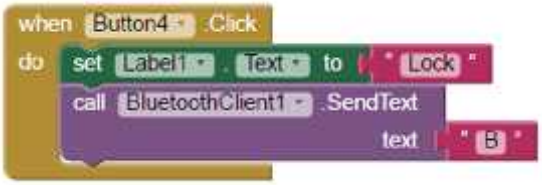


Gambar 32. *Blocks* Tombol *Disconnect*

Ketika “Button3” / tombol *Disconnect* di tekan maka “BluetoothClient1” akan memutuskan koneksi dan “Button3” / tombol *Disconnect* akan berubah warna menjadi merah.

Gambar 33. *Blocks* Tombol *Unlock*

Ketika “Button5” / tombol *Unlock* di tekan maka “BluetoothClient1” akan mengirimkan data berupa karakter “A” dan text pada “Label1” akan berubah menjadi “Unlock”.



Gambar 34. *Blocks* Tombol *Lock*

Ketika “Button4” / tombol *Lock* di tekan maka “BluetoothClient1” akan mengirimkan data berupa karakter “B” dan text pada “Label1” akan berubah menjadi “Lock”.

Gambar 35. *Blocks* Tombol *Garage Door Controller*

Ketika “Button1” / tombol *Garage Door Controller* di tekan maka akan mengarahkan ke “Screen2” yang akan menampilkan tentang aplikasi.

Gambar 36. *Blocks* Tombol *Exit*

Ketika “Button6” / tombol *Exit* di tekan aplikasi akan keluar.

Gambar 37. *Blocks* Tombol *Help*

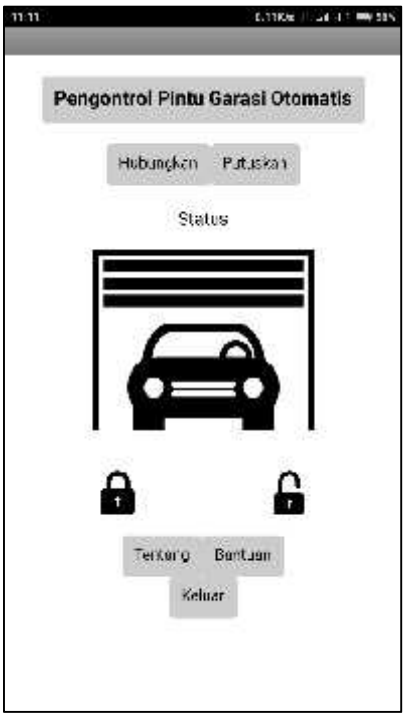
Tombol *Help* berada pada “Screen2” / *form* tentang dimana terdapat tentang aplikasi, ketika “Button1” / tombol *Help* di tekan maka akan mengarahkan ke “Screen3” / *form* bantuan yang berisi bantuan cara menggunakan aplikasi sedangkan ketika tombol navigasi *back* pada *smartphone* Android di tekan maka akan kembali ke “Screen1” / *form* utama.

Gambar 38. *Blocks* Tombol *Back*

Blocks ini terdapat pada “Screen3” / *form* bantuan ketika tombol navigasi *back* pada *smartphone* Android di tekan maka akan kembali ke “Screen2” / *form* tentang.

3) Pembahasan *Interface* / Antarmuka Program

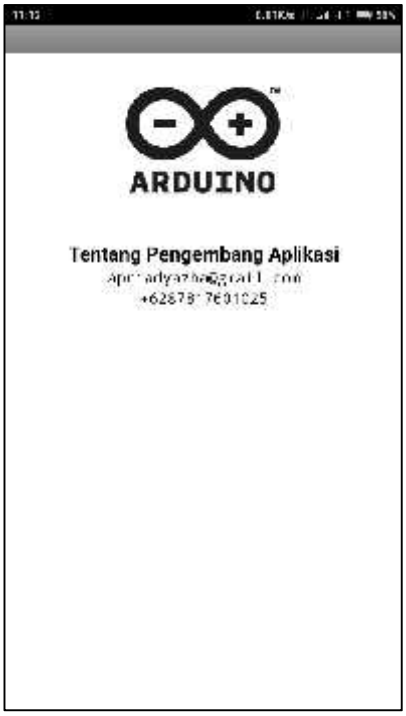
a. Tampilan Utama



Gambar 39. Tampilan Utama

Pada tampilan utama terdapat dua buah tombol utama yaitu tombol Buka dan Tutup yang digunakan untuk membuka dan menutup pintu garasi, sedangkan tombol Hubungkan dan Putuskan digunakan untuk melakukan koneksi ke perangkat yang terhubung serta terdapat tombol Keluar untuk keluar dan tombol untuk mengakses tentang.

b. Tampilan Tentang



Gambar 40. Tampilan Tentang

Pada tampilan tentang terdapat informasi mengenai kontak *programmer*.

c. Tampilan Bantuan

Gambar 41. Tampilan Bantuan

Pada tampilan bantuan terdapat cara penggunaan aplikasi, serta berisi tahapan-tahapan dalam melakukan koneksi ke parangkat yang terhubung.

4) Pembahasan Hasil Respon Pengguna

Pada sub bab ini penulis akan membahas mengenai hasil respon pengguna (hasil kuisioner) yang bertujuan untuk mengetahui tanggapan dari pengguna yang menguji langsung kenerja “Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonic HC-SR04”.

a. Kuisioner

Kuisioner merupakan tahapan yang menunjukan tentang pernyataan apa saja yang diajukan kepada responden terhadap "Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonic HC-SR04".

Tabel 3. Kuisioner

No.	Pernyataan	Alternatif Jawaban				
		Sangat Setuju	**Setuju**	**Cukup Setuju**	**Kurang Setuju**	**Tidak Setuju**
1.	Semua layout dan halaman dapat di tampilkan.					
2.	Semua tombol aplikasi dapat berfungsi dengan baik.					
3.	Pengoperasian aplikasi mudah dipahamai.					
4.	Aplikasi dapat dijalankan dengan baik.					

*)Keterangan bobot penilaian :

Keterangan	Nilai
Sangat Setuju	5
Setuju	4
Cukup Setuju	3
Kurang Setuju	2
Tidak Setuju	1

b. Hasil Kuisioner

Hasil kuisioner merupakan hasil dari perhitungan dari penilaian responden terhadap “Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonik HC-SR04”.

Tabel 4. Hasil Kuisioner

No	Nama	Item Pertanyaan				Jumlah Skor	Rata-Rata
		1	2	3	4		
1.	Edy Kurniawan	4	5	5	5	19	4,75
2.	Dwi Rahayu A.	5	5	4	4	18	4,5
3.	Fendy Ibrahim	4	4	4	5	17	4,25
4.	Heri Nanda	4	4	5	5	18	4,5
5.	Bely Julianto	4	5	4	5	18	4,5
6.	Haveria	4	4	4	5	17	4,25
7.	Ibrahim Thoimi	5	4	5	4	18	4,5
8.	M. Ziaurrahman	4	4	4	4	16	4
9.	Bryan Rofson	5	4	4	4	17	4,25
10.	Ginanjar Basuki	5	5	4	4	18	4,5
Total						176	4,4

Untuk mengetahui jumlah skor kriterium berdasarkan tabel 4, dapat dihitung menggunakan rumus sebagai berikut :

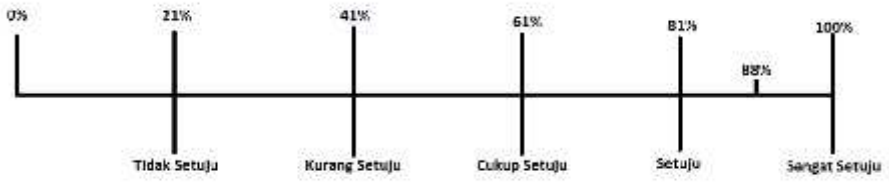
Jumlah skor kriterium = skor tertinggi tiap item x jumlah pertanyaan x jumlah responden

Jumlah skor kriterium = 5 x 4 x 10 = 200

Berdasarkan tabel 4, jumlah total skor adalah 176 dan perhitungan jumlah skor kriterium adalah 200, dengan demikian “Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonik HC-SR04” menurut persepsi 10 responden, yaitu 176 / 200 x 100% = 88% tergolong

“Sangat Setuju” bahwa sistem ini layak di implementasikan dan digunakan.

Persentase berdasarkan kelompok responden dapat dilihat pada Gambar 42.



Gambar 42. Persentase Hasil Kuisioner

Keterangan :

Kriteria Interpresentasi Skor

Angka 0% - 20%	= Tidak Setuju
Angka 21% - 40%	= Kurang Setuju
Angka 41% - 60%	= Cukup Setuju
Angka 61% - 80%	= Setuju
Angka 81% - 100%	= Sangat Setuju

BAB V
PENUTUP

A. Kesimpulan

Berdasarkan hasil pemaparan pada bab-bab sebelumnya seperti analisis, desain sistem, implementasi sistem serta pembahasan, maka penulis dapat menarik kesimpulan pada Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultrasonik HC-SR04 sebagai berikut :

1. Rancang bangun pintu garasi otomatis menggunakan arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 ini dibangun menggunakan *board* Arduino, sensor ultrasonik HC-SR04 sebagai pendeteksi objek, servo sebagai penggerak pintu garasi dan bluetooth HC-05 sebagai perangkat kerasnya.
2. Rancang bangun pintu garasi otomatis menggunakan arduino UNO R3 dengan memanfaatkan sensor ultrasonik HC-SR04 ini dapat dikonfigurasikan menggunakan aplikasi kontroler yang ada pada *smartphone* dengan memanfaatkan koneksi bluetooth.
3. Berdasarkan perhitungan persentase kelompok responden sebanyak 10 responden dengan hasil persentase 88% yang berarti “Sangat Setuju” sistem ini layak digunakan dan di implementasikan.

B. Saran

Adapun saran yang diberikan penulis untuk kedepannya, penulis mengharapkan sebagai berikut :

1. Pada pengembangan berikutnya diharapkan dapat di implementasikan pada kondisi sebenarnya.
2. Dibutuhkan tambahan sensor agar objek yang dikenali benar-benar mobil dan bukan benda lain.
3. Indikator objek hanya berupa LED perlu ditambahkan notifikasi pada aplikasi *smarthphone* Android agar ketika garasi terbuka pemilik mengetahui keadaan garasi.
4. Dibutuhkan tambahan daya untuk motor servo.
5. Faktor keamanan harus ditingkatkan.

DAFTAR PUSTAKA

Jatmika, Hafidz Fadillah. 2015. “Perancangan Pengotrolan Gerbang Menggunakan Bluetooth Dengan Interface Android Berbasis Arduino UNO Pada PT. NIPSEA Paint and Chemicals”. Skripsi Jurusan Sistem Komputer STMIK Rajarja, Tanggerang.

Jogiyanto, HM. 2001. *Analisis Perancangan Sistem Informasi*. Andi Yogyakarta, Yogyakarta.

Kadir, Abdul. 2013. *Panduan Praktis Mempelajari Aplikasi Mikrokontroler dan Pemrogramannya Menggunakan Arduino*. Andi, Yogyakarta.

Kadir, Abdul. 2015. *From Zero to A Pro Arduino*. Andi, Yogyakarta.

Mantra, Brata Abi. 2006. “Simulasi Pintu Garasi Mobil Otomatis Berbasis PLC (Programmable Logic Control)”. Tugas Akhir Fakultas Teknik Universitas Negeri Semarang.

Maryanto, Hendra. 2010. “Pembuatan Prototipe Pintu Otomatis Satu Arah Berbasis Mikrokontroler ATMega 8535 Menggunakan Double IR”. Tugas Akhir Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Sebelas Maret Surakarta.

Mohammad Thoif. 2006. “Pengendali Pintu Gerbang dan Garasi Secara Otomatis”. Tugas Akhir Fakultas Teknik Universitas Negeri Semarang.

Reza, Tubagus. 2007. “Pintu Geser Otomatis”. Skripsi Universitas Gunadarma.

Prasetiyo, Ahmad Fajar. 2011. App Inventor untuk Pemula, Surya University.

Pressman, Roger S. 2010. *Software Enginering A Practitioner’s Approach*. New York.

Pressman, Roger S. 2012. *Rekayasa Perangkat Lunak (Pendekatan Praktisi) Edisi 7, Buku 1*. Andi, Yogyakarta.

Saftari, Firmansyah. 2015. *Proyek Robotika Keren dengan Arduino*. Elex Media Komputindo. Jakarta.

Santoso, Hari. 2015. Panduan Praktis Arduino Untuk Pemula. Trenggaleg.

Sayudhi, Taufiq Dwi Septian. 2010. *Buku Pintar Robotika*. Andi Yogyakarta, Yogyakarta.

STMIK Palangkaraya. 2014. *Pedoman Penulisan Proposal dan Skripsi Teknik Informatika*. Palangka Raya.

Sugiyono. 2009. *Metode Penelitian Kuantitaif, Kualitatif dan R&D*. Alfhabeta, Jakarta

Triatna, Asep Tantan. 2013. *"Studi Deskriptif Analisis"*. Bandung.

BERITA ACARA EKSPERIMEN PROGRAM

Pada hari ini ..... tanggal ..... bulan ..... tahun ..... telah dilaksanakan uji coba program dalam penulisan skripsi :

Nama : Apriady
NIM : C1255201020
Judul : Rancang Bangun Pintu Garasi Otomatis Menggunakan Arduino UNO R3 Dengan Memanfaatkan Sensor Ultasonik HC-SR04

Poin-poin yang di eksperimen yaitu :

No.	Testing Evaluasi	Status *)	Keterangan
1.	Tes Aplikasi		
2.	Tes Sensor 1		
3.	Tes Sensor 2		
4.	Tes Membuka Pintu		
5.	Tes Menutup Pintu		

Keterangan :

*) Diisi ***Ya*** jika program berhasil dijalankan sesuai permasalahan dan ***Tidak*** jika program tidak berhasil dijalankan susuai permasalahan.

Demikian berita acara ini dibuat dengan sebenar-benarnya untuk dapat dipergunakan sebagai persyaratan ujian skripsi.

Palangkaraya, ...

Yang Menguji, Mahasiswa Yang Diuji,

…… ……

KUISIONER RANCANG BANGUN PINTU GARASI OTOMATIS MENGGUNAKAN ARDUINO UNO R3 DENGAN MEMANFAATKAN SENSOR ULTRASONIK HC-SR04

Nama : Merk/ Tipe *Smartphone* :
Pekerjaan : OS *Smartphone* :

No.	Pertanyaan	Alternatif Jawaban				
		Sangat Setuju	**Setuju**	**Cukup Setuju**	**Kurang Setuju**	**Tidak Setuju**
1.	Semua layout dan halaman dapat di tampilkan.					
2.	Semua tombol aplikasi dapat berfungsi dengan baik.					
3.	Pengoperasian aplikasi mudah dipahamai.					
4.	Aplikasi dapat dijalankan dengan baik.					

Keterangan :

*) Berilah tanda checklist ( ) pada kolom jawaban yang sesuai dengan pendapat anda.

Palangkaraya, …….
Responden,

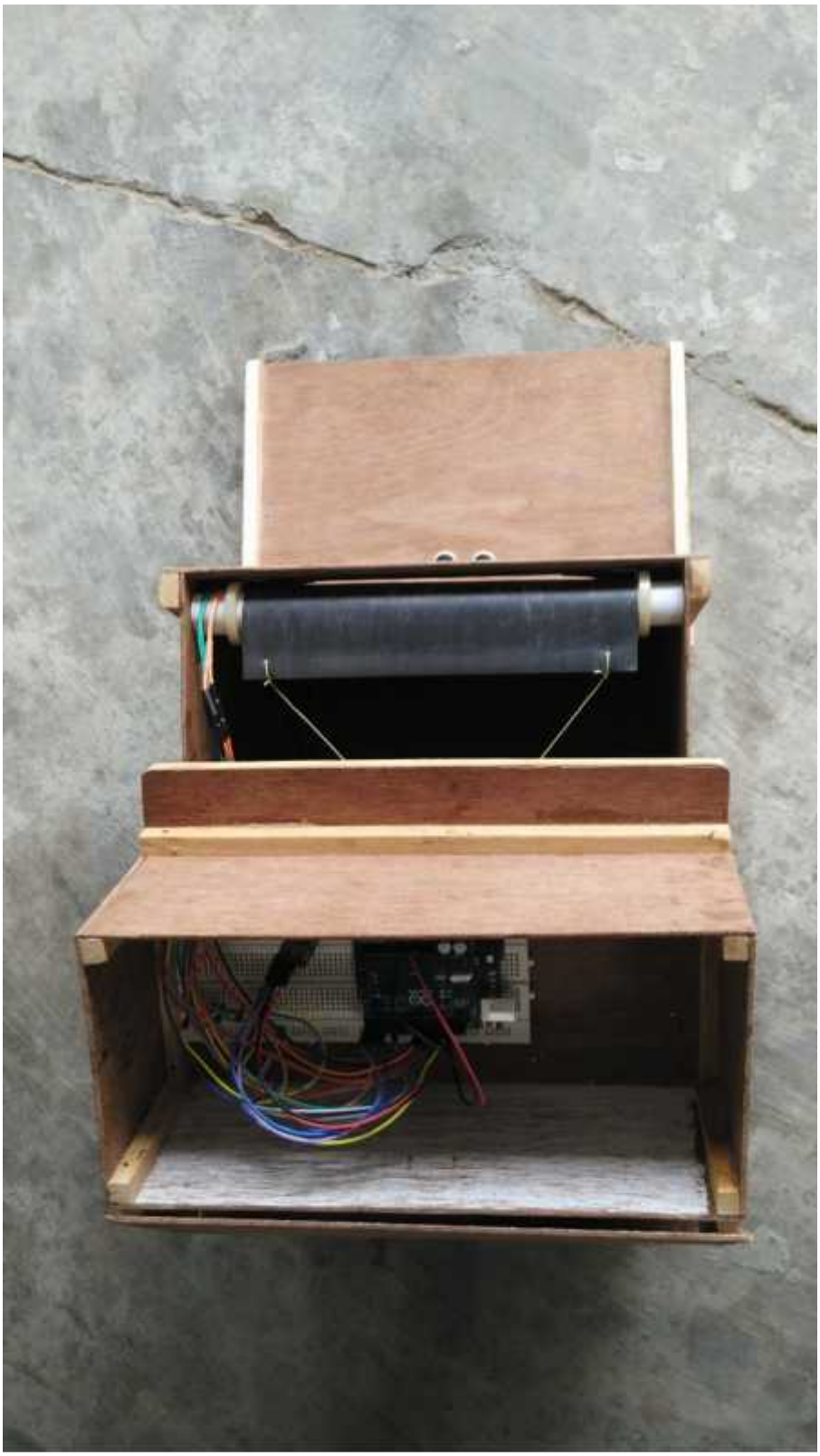
(……)

Prototype 1 Pembuatan Garasi

Prototype 2 Pembuatan Tempat Rangkaian Sistem Komponen

Prototype 3 Pemasangan Pintu Garasi, LED dan Sensor Ultrasonik

Prototype 4 Pemasangan Sensor Ultrasonik dan Sistem Komponen



Prototype 5 Pemasangan Garasi dan Tempat Sistem Komponen

Prototype 6 Pemasangan Servo dan Sistem Penggerak Pintu Garasi